

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

Факультет електроніки
(повна назва інституту/факультету)

Кафедра мікроелектроніки
(повна назва кафедри)

«На правах рукопису»
УДК _____

«До захисту допущено»

Завідувач кафедри

_____ (підпис)

_____ (ініціали, прізвище)

“ _____ ” _____ 20__ р.

Магістерська дисертація

на здобуття ступеня магістра

зі спеціальності 153 “Мікро- та наносистемна техніка”
(код і назва)

на тему: Інтелектуальна система моніторингу та клімат контролю приміщень

Виконав (-ла): студент (-ка) 6 курсу, групи ДП-82мп
(шифр групи)

Крисенко Павло Ігорович
(прізвище, ім'я, по батькові) (підпис)

Науковий керівник перший проректор, д.т.н., проф., Якименко Ю.І.
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Консультант з нормоконтролю проф., к.т.н., доц. Орлов А.Т.

Консультант з інформаційних питань доц., к.т.н. Діденко Ю.В. _____

Рецензент
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Засвідчую, що у цій магістерській дисертації
немає запозичень з праць інших авторів без
відповідних посилань. Студент

_____ (підпис)

Київ – 20__ року

Реферат

81 сторінка, 8 розділів, 27 ілюстрацій, 31 таблиця, 17 джерела за переліком посилань

В наш час швидко розвивається галузь штучного інтелекту широко використовується у повсякденному житті. Цьому сприяв розвиток електронно обчислювальних машин, активне інвестування фірм-гігантів, таких як Google, Amazon, Samsung, Huawei та інші, у науково дослідницькі компанії та ентузіастів. Також існує попит звичайних користувачів до автоматизації їх життя, моніторингу робочих приміщень та лабораторій.

Тому головне завдання дисертаційної роботи є розробка інтелектуальної системи моніторингу та клімат контролю приміщень. Щоб досягти цієї мети були виконані такі завдання:

- розгляд варіантів реалізації на сучасних обчислювальних системах
- розгляд технологій бездротового зв'язку та протоколи бездротової передачі інформації
- був зроблений огляд штучних нейронних мереж, аналіз структури сучасних нейронних мереж та була обрана архітектура, котра може вирішити задачу
- збір даних для навчання штучної нейронної мережі з обраною архітектурою
- було проведено навчання штучної нейронної мережі на зібраних даних
- реалізована архітектура клієнту та серверу для обміну між ними інформацією
- зроблені висновки о можливості створення та використання інтелектуальної системи моніторингу та клімат контролю приміщень

Об'єктом дослідження є системи моніторингу та клімат контролю приміщень.

Було досліджено метод прогнозування температурних та вологих явищ за допомогою штучної рекурентної нейронної мережі з ячейкою довгої короткочасної пам'яті.

Ключові слова: Raspberry Pi, Arduino, PSoC, сокети, штучна нейронна мережа, рекурентна нейронна мережа, довга короткочасна пам'ять

Abstract

80 pages, 8 sections, 27 illustrations, 31 tables, 17 reference lists

Nowadays, the rapidly developing field of artificial intelligence is widely used in everyday life. This was facilitated by the development of electronic computers and the active investment of giants such as Google, Amazon, Samsung, Huawei and others in research companies and enthusiasts. There is also demand from ordinary users to automate their lives, monitor workplaces and labs.

Therefore, the main task of the dissertation is to develop an intelligent monitoring system and climate control of the premises. To achieve this, the following tasks have been accomplished:

- consideration of implementation options on modern computing systems
- review of wireless technology and wireless protocols
- a review of artificial neural networks, analysis of the structure of modern neural networks and the architecture that can solve the problem
- collecting data to train an artificial neural network with the chosen architecture
- was trained artificial neural network on the collected data
- implemented client and server architecture to exchange information between them
- conclusions were drawn about the possibility of creating and using an intelligent monitoring system and climate control of premises

The subject of the study is the monitoring systems and climate control of the premises.

A method for predicting temperature and humidity phenomena using an artificial recurrent neural network with a cell of long-term short-term memory was investigated.

Keywords: Raspberry Pi, Arduino, PSoC, sockets, artificial neural network, recurrent neural network, long short-term memory

Зміст

Вступ.....	8
1. Системи на кристалі.....	9
1.1. PSOC.....	9
1.2. Arduino.....	12
1.3. Raspberry Pi.....	15
1.4. Вибір обчислювальної системи.....	18
2. Програмний інтерфейс для передачі інформації.....	20
2.1. Сокети.....	20
3. Датчики температури.....	22
3.1. Датчик температури та вологості DHT11.....	22
3.2. Датчик температури та вологості DHT22.....	24
3.3. Вибір робочого датчику.....	27

4. Штучні нейронної мережі.....	29
4.1. Історія розвитку штучних нейронних мереж.....	29
4.2. Задачі, котрі вирішують штучні нейронні мереж.....	29
4.3. Штучний нейрон.....	31
4.4. Архітектури штучних нейронних мереж.....	33
4.5. Довга короткочасна пам'ять.....	37
4.6. Навчання штучної нейронної мережі.....	40
4.7. Алгоритм зворотного поширення помилки.....	44
4.8. Проблеми штучних нейронних мереж.....	46
5. Процес навчання штучної нейронної мережі.....	48
5.1. Архітектура обраної штучної нейронної мережі.....	48
5.2. Збір даних для навчання штучної нейронної мережі.....	49

5.3. Етап навчання.....	51
6.Архітектура прототипу інтелектуальної системи датчиків.....	55
6.1 Архітектура серверу.....	55
6.2 Архітектура клієнту.....	57
7. Збір інтелектуальної системи.....	59
7.1. Підключення сенсору DHT22 до Raspberry Pi.....	59
8. Розробка стартап проекту.....	64
8.1. Опис ідеї стартап проект.....	64
8.2. Технологічний аудит ідеї проекту.....	65
8.3. Аналіз ринкових можливостей запуску стартап-проекту.....	66
8.4. Розроблення ринкової стратегії стартап проекту.....	74
8.5. Розроблення маркетингової програми стартап-проекту.....	76

Висновки.....	
78	
Список літератури.....	79

Перечень скорочень

ЦАП - цифро аналоговий перетворювач

АЦП - аналого цифровий перетворювач

ФНЧ - фільтр низких частот

ТТЛ - транзисторно транзисторна логіка

ЕОМ - електронно обчислювальна машина

РНМ - рекурентна нейронна мережа

ДКЧП - довга короткочасна пам'ять

Вступ

В даному звіті буде викладена інформація про проходження переддипломної практики на кафедрі мікроелектроніки факультету електроніки. За темою моєї дипломної роботи, а саме «Інтелектуальна система датчиків» була проведена дослідницька робота. Був обраний одноплатний комп'ютер, а саме Raspberry Pi 3B+, в якості основного елементу, до якого будуть під'єднані сенсори, також була обрана технологія для бездротової локальної мережі, для зв'язку між основними елементами системи та пристроєм, до якого надходитеме вся інформація, та на якому відбуватиметься обробка її. В якості програмного інтерфейсу для забезпечення передачі інформації були обрані сокети. В якості сенсора вологи та температури буде використовуватись датчик DHT22. Інтелектуальність системи буде полягає в тому, що вона здатна прогнозувати значення температури та вологості у приміщенні, в якості математичної моделі, буде використовуватися рекурентна нейронна мережа з ячейкою довгої короткочасної пам'яті. Навчання такої нейронної мережі буде відбуватися на графічному процесорі Nvidia GTX 1050 TI. Універсальність системи полягає в можливості роботи за нею з будь-якої операційної системи.

1. Системи на кристалі

1.1. PSoC

PSoC - система, яка вміщує функціональні складові системи на одному чипі, й може бути запрограмовано користувачем.[10] Цей пристрій відрізняється від мікроконтролерів процесорним ядром, PSoC має в собі матрицю аналогових та цифрових блоків. Ця система була розроблена компанією Cypress Semiconductor.[11] Так як такі цифрові та аналогові блоки можуть бути запрограмовані, ми маємо можливість створення на основі цього такого функціоналу, як АЦП, ЦАП, ФНЧ, компаратор, тощо.



Рис. 1.1.1. PSoC 5

На даний момент було розроблено чотири покоління систем.

Таблиця 1.1.1 Порівняння характеристик PSoC 1 та PSoC 3[12]

	PSoC 1	PSoC 3
Ядро	8-ми бітний M8C core з частотою 24 MHz, 4 MIPS	8-bit 8051 core з частотою 67 MHz, 33 MIPS
Пам'ять	SRAM: 256 b - 2 KB Flash: 4 - 32 KB	SRAM: 3 - 8 KB Flash: 8 - 64 KB
Інтерфейси, які підтримують ся	UART, FS USB 2.0, I ² C, SPI,	FS USB 2.0, I ² S, CAN, I ² C, SPI, UART, LIN
АЦП/ЦАП	До двуканальний ЦАП (від 6 до 8-bit) 1 дельта-сігма АЦП (від 6 до 14-bit) 131 ksps 8-bit;	До 4 чотирьох ЦАП (8-bit) 1 дельта-сігма АЦП (від 8 до 20-bit) 192 ksps @ 12-bit;
Кількість вводів/виводів	До 64	До 72
Живлення	від 1.7 В до 5.25 В Споживання: в активному режимі: 2 мА, у сплячому режимі: 3 мкА Бездіяльний режим: відсутній	від 0.5 В до 5.5 В Споживання: у активному режимі: 1,2 мА, у сплячому режимі: 1 мкА, у бездіяльному режимі: 200 нА

Таблиця 1.1.2 Порівняння характеристик PSoC 4 та PSoC 5LP[12]

	PSoC 4	PSoC 5LP
--	--------	----------

Ядро	32-ух бітний ARM Cortex-M0 з частотою 48 MHz, 65 MIPS	32-ух бітний ARM Cortex-M0 з частотою 48 MHz, 84 MIPS
Пам'ять	Flash: 16 - 32 KB SRAM: 2 - 4 KB	SRAM: 8 - 64 KB Flash: 32 - 256 KB
Інтерфейси, що підтримуються	UART, I2C, SPI	I2C, FS USB
ЦАП/АЦП	1 Msps @ 12-bit; 1 SAR АЦП (12-bit) До 2-ух ЦАП (з 7 до 8-bit)	1 Msps @ 12-bit; 2 SAR АЦП (12-bit) 1 дельта-сігма-дельта АЦП (12-bit) 192 ksps @ 12-bit До 4-ох ЦАП
Кількість вводів/виводів	До 36	
Живлення	від 1.71 В до 5.5 В Споживання: у активному режимі: 1.6 мА, у сплячому режимі: 1.3 мкА, у бездіяльному режимі: 150 нА	від 2.7 В до 5.5 В Споживання: у активному режимі: 1.6 мА, у сплячому режимі: 1.3 мкА, у бездіяльному режимі: 150 нА

Як видно з табл. 1.1.1 та табл. 1.1.2 більш сучасна та потужна система на кристалі PSoC 5LP, яка показана на рис.1.1.1. Але, ця система має значний недолік, а саме - відсутні модулі bluetooth та Wi-Fi "із коробки"[12], але, деякі виробники електроніки виробляють набори Wi-Fi модулів, такі як Wi-Fi Serial Transceiver Module w& ESP8266, серія Wi-Fi модулів Wi-Fi Secure MCU RS14100-SB00 від компанії Redpine Signals[13].

1.2. Arduino

Arduino - це апаратно-обчислювальна платформа, яка складається з таких компонентів: плата вводу й виводу та середовище розробки. Arduino добре

підходить для розробки різноманітних електронних пристроїв, як для новачків, такі для професійних розробників. Ця система має велику кількість прихильників у всьому світі за рахунок простої мови програмування, відкритої архітектури та коду. Особливість цієї платформи є у тому, що , що вона програмується без программатору, лише через USB. За використанням ардуіно ми можемо вийти за рамки віртуального світу у реальний, з таким різноманіттям датчиків, які ми можемо підключити до нашої плати ардуіно. Виходячи з цього, ми, як розробники, можемо керувати різноманітними двигунами, освітленням та іншими функціональними блоками, отримувати інформацію про зовнішнє середовище, таку як тиск, температура, вологість, освітлення, тощо. Також ця платформа має змогу взаємодіяти з таким ПЗ на персональному комп'ютері, як наприклад, Processing, MaxMSP. Плата Arduino складається з програмованих елементів, інтеграції з іншими схемами та мікроконтролеру. Більшість ардуіно мають лінійний регулятор напруги.

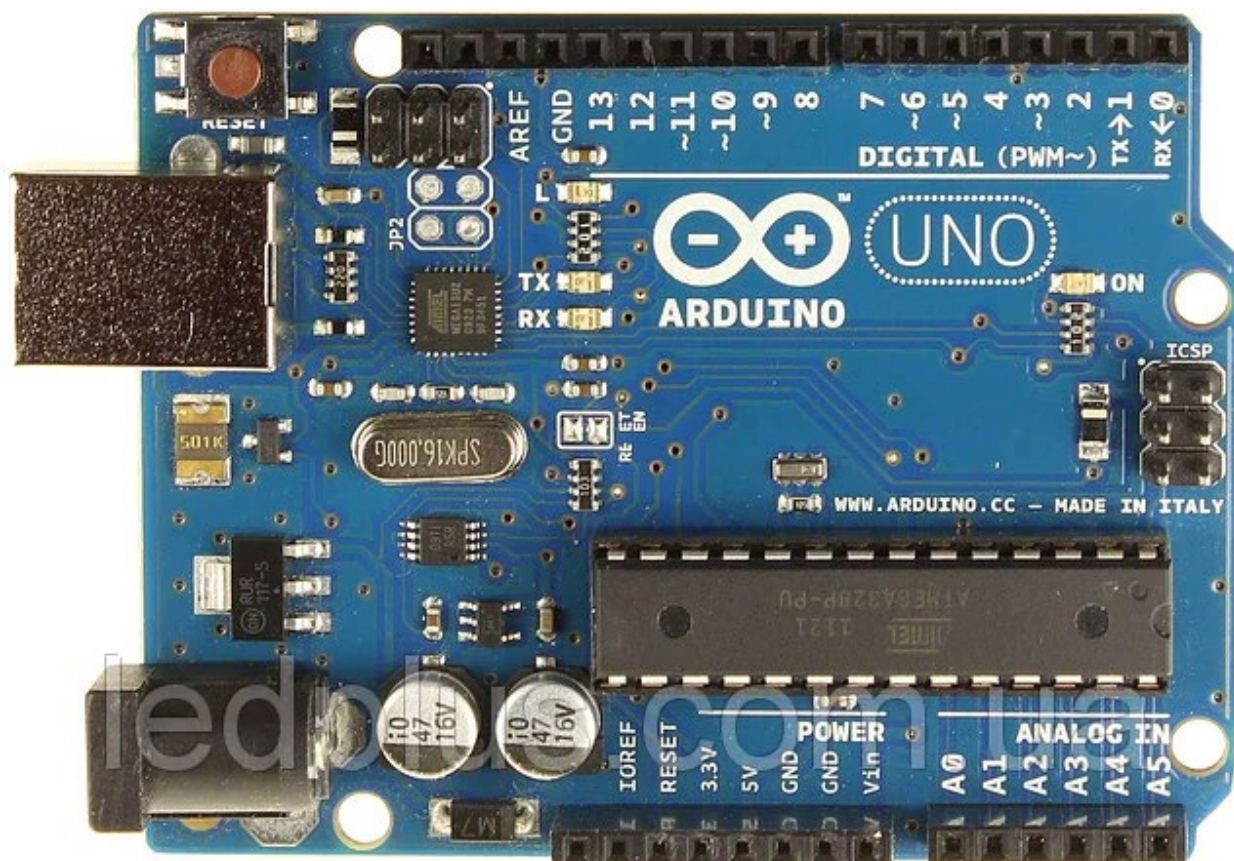


Рис. 1.2.1. Arduino Uno R3

ння виконується на частотах 8 або 16 МГц за допомогою кварцевого резонатору (прилад, який заснований на явищі пьезокристалічного ефекту та явищі механічного резонансу, в якому ці явища використовуються для створення високочастотного (добротність це властивість коливальних систем, за рахунок якої визначається смуга резонансу і визначає в скільки разів запас енергії в системі більший, ніж втрати енергії за одне коливання), що є резонансним елементом електронної схеми. Спочатку в мікроконтролер прошивається завантажувач(це програма. яка відповідає за завантаження виконавчих файлів та запуск процесів) , який носить назву BootLoader, саме тому зовнішній програматор не використовується. В платі Arduino(рис.1.2.1) міститься інвертуюча схема, що використовується щоб конвертувати рівні сигналів RS-232 (recommended standart 232,це фізичний рівень для асинхронного інтерфейсу) в рівні транзисторно-транзисторної логіки цифрових логічних мікросхем, які побудовані на основі біполярних транзисторів та резисторів.).Інтегроване

середовище розробки Arduino - це додаток, що був написаний на мові програмування Java, який включає в собі редактор для редагування коду, компілятор та модуль для прошивки плати. Мова програмування Arduino має велику схожість іншу мову програмування C++, доповнену всілякими бібліотеками. Обробка програм здійснюється препроцесором, а компілювання відбувається за допомогою AVR-GCC

Зараз існує багато модифікацій плат Arduino тому, треба зробити якісний аналіз існуючих модифікацій.

Таблиця 1.2.1 Порівняння основних моделей Arduino[13]

	Nano	Uno	Mega	Leonardo
Процесор	ATmega168 чи ATmega328	ATmega328P	ATmega1280	Atmega32u4
Напруга живлення	5 В	5 В	5 В	5 В
Флеш-пам'ять, КБ	16/32	32	128	32
EEPROM, КБ	0.5/1	1	4	1
SRAM, КБ	1/2	2	8	2
Двійкові входи/виходи	14	14	54	14
Аналогові входи	8	6	16	12

Як видно з табл. 1.2.1 найбільш потужна модель - Mega. Але, так як і у випадку з PSoC, плати Arduino не мають Wi-Fi модулі з коробки, тому доведеться окремо дозакупати модулі. Найпоширенішими модулями Wi-Fi для Arduino є сімейство модулів wi-fi jammer esp8266, ціна яких починається в 2 доларів. Недоліками такої платформи є низька швидкодія, малий об'єм пам'яті, обмежений функціонал.

1.3. Raspberry Pi

Raspberry Pi це одноплатний комп'ютер, який побудований на системі на чипі (SoC) Broadcom BCM2835, який включає в себе процесор ARM з тактовою частотою 700 мегагерц, процесор для обробки графіки VideoCore IV, і операційний запам'ятовувальний пристрій на 256 або 512 мегабайтів[4]. Жорсткий диск на Raspberry Pi відсутній, замість нього в наявності є SD карта. Така апаратне забезпечення дозволяє програвати відео форматі H.264 в роздільній здатності 1080, та запускати такі комп'ютерні ігри як Quake III Arena.

Ініціатор цього проекту це Raspberry Pi Foundation, британський благодійний фонд. У планах було використовувати цей комп'ютер як майданчик для дітей, що хочуть займатися програмуванням, але знайшов прихильників і в інших сферах — наприклад, на основі Raspberry Pi розробляють домашні медіацентри. Найдешевший одноплатний комп'ютер продається без корпусу та виглядає як плата розміром зі звичайну кредитну карту. Плата важить лише сорок п'ять грамів. У Raspberry Pi використовується, як зазначалося вище, 700-мегагерцовий процесор з архітектурою ARM; в наявності роз'єм для навушників та слот під карти пам'яті. Старша та молодша моделі Raspberry Pi мають різний об'єм оперативної пам'яті (512 й 256 мегабайтів відповідно) та кількістю USB-портів (два проти одного). Окрім цього, у старшої моделі в наявності Ethernet роз'єм 10/100, але молодша на треть більш енергоефективна.

Модель B Raspberry Pi надійшла у продаж у 2012 році, в кінці лютого, за привабливою ціною у 35 доларів США. Через рік у продажі в Європі з'явилась молодша модель

одноплатного комп'ютера вартістю на доларів дешевша, ніж старша модель, тобто 25 доларів США.[14] Комп'ютери Raspberry Pi продаються у

представників Element14 та RS Components. Від початку 2012 року були реалізовані більше мільйона одноплатних комп'ютерів.

Однією з найцікавіших особливостей Raspberry Pi є наявність портів GPIO (general purpose input / output). Завдяки цьому "малиновий" комп'ютер можна використовувати для управління різними пристроями. У моделі «В» плати присутні 26 портів, а в моделі «В +» і «2» - 40 портів GPIO. Контакти GPIO використовуються для отримання інформації від різноманітних датчиків, як приклад такими датчиками можуть бути: датчик ІЧ, відео, температури, орієнтації, прискорення, світлу, вологості, тиску, вібрації, вогню, диму. Також ці роз'єми можуть використовуватися для безпосереднього керування двигуном на постійному струмі, аудіо, відео, ЖК-дісплеєм, світлодіодами, перемикачами, реле.

Raspberry Pi працює в основному на операційних системах, заснованих на Linux ядрі. Запуск Windows можливий завдяки засобам віртуалізації таким, як XenDesktop. ARM11 заснований на 6 версії ARM, на якому кілька популярних версій Linux більше не запускається. Для установки операційних систем існує інструмент NOOBS

Вбудований пристрій для читання карт пам'яті гарантовано працює з більшістю SD-карт об'ємом до 32 Гбайт. Завантажуватися Raspberry Pi вміє тільки з карток SD. Якщо точніше, то сама ОС може розташовуватися на USB-накопичувачі, але ось завантажувач завжди повинен бути на SD. Кнопка включення і скидання немає - пристрій сам включається при подачі живлення. Харчується Raspberry Pi від порту micro-USB або з пари виділених висновків GPIO. Для Model A рекомендується джерело на 5 В і 500-700 мА, а для Model B на 5 В і 700-1200 мА. Тобто порту USB 3.0 або зарядного пристрою для телефону має вистачити, хоча краще підібрати більш стабільне джерело живлення. Самі плати споживають трохи менше, але частина енергії потрібно для роботи підключених до USB-портів пристроїв. Альтернативний варіант - живлення від підключеного до плати USB-хаба з окремим БП або акумуляторів,

але це не найкраще рішення. До речі, контролер Ethernet в Model B теж «висить» на шині USB. Індикація мінімальна - на платі розпаяно п'ять світлодіодів. Три з них вказують на активність і режим роботи Ethernet, а ще два сигналізують про наявність харчування і роботі з SD-картою.

Для Raspberry Pi випущено спеціалізований дистрибутив Linux, Raspbian OS (заснований на дистрибутиві Debian) і магазин застосунків Pi Store, де є як платні, так і безплатні програми. Серед іншого Raspberry Pi використовує веб-браузер Iceweasel та KOffice, які постачаються в комплекті. Офіційна мова програмування для навчання — Python. Лише останні три модифікації Raspberry Pi мають вбудований Wi-Fi модуль, тому має сенс розглядати лише їх

Таблиця 1.3.1 Порівняння модифікацій Raspberry Pi

	3B	3B+	4B
Процесор	ARM Cortex-A53 x64	ARM Cortex-A53 x64	ARM Cortex-A72 x64
Частота	1,2 ГГц	1,4 ГГц	1,5 ГГц
Кількість ядер	4	4	4
ОЗП, ГБ	1	1	1-4
GPIO	40 пінів	40 пінів	40 пінів
USB	4 порти	4 порти	4 порти

Ethernet	Так	Так	Так
Wi-Fi	802.11n	двосмуговий 2.4/5 ГГц 802.11b/g/n/ac Wi-Fi	2,4 ГГц та 5,0 ГГц IEEE 802.11ac

Всі три модулі мають вбудований Wi-Fi модуль та потужний процесор, що робить їх гарним варіантом для використання, в якості основної системи.

1.4 Вибір обчислювальної системи

Взявши до уваги всі недоліки та переваги, я зупинився на Raspberry Pi 3B+, тому що:

- має велику популярність у ентузіастів
- підтримка з боку спільноти ентузіастів та виробника, які завжди допоможуть з проблемами платформи
- повноцінний комп'ютер, який обладнаний 4-ма USB виходами,

HDMI.

- 64-бітний чотирьохядерний процесор
- великий вибір операційних систем
- розміри комп'ютеру можна порівняти з розмірами кредитної карти
- є можливість розгону центрального процесору, якщо не буде вистачати його обчислювальної здатності
- невисока ціна, яка приблизно дорівнює тридцяти п'яти долларам Сполучених Штатів Америки
- відсутність шуму під час роботи, так як використовується система пасивного охолодження
- низький рівень енергоспоживання

- вбудований Wi-Fi модуль

PSoC 5LP також гарний варіант у якості основної системи, але якщо врахувати ціну самої плати і Wi-Fi модуля, то вона буде перевищувати ціну Raspberry Pi 3B+ на 20 доларів. Також зручність обраного одноплатного комп'ютера в можливості вести розробку на мові Python.

2. Програмний інтерфейс для передачі інформації

2.1 Сокети

Сокет це назва програмного інтерфейсу який використовується для передачі даних між системними процесами. Процеси при такому обміні даними можуть виконуватися на одній електронно обчислювальній машині й, на різних електронно обчислювальних машинах, які пов'язані між собою інтернет мережею. Сокет - це такий абстрактний об'єкт, що дає змогу мати кінцеву точку з'єднання процесів. Кожен процес створює серверний сокет і забіндити його до деякого порту операційної системи (але *nix процеси без суперправ не будуть мати доступ до портів, які менше 1024). Процес, який слухатиме, зазвичай буде знаходитися очікування, себто очікувати появи нового з'єднання на порту. При цьому зберігатиметься можливість перевіряти наявність з'єднань в цей момент, встановлювати час від'єднання для операцій тощо[1]. Кожен сокет обов'язково матиме свою унікальну адресу. Операційні системи сімейства юнікс підтримують багато різних типів адрес для сокетів, але обов'язковими являються unix-адреса та inet-адреса. Якщо забіндити сокет до адреси юнікс, то створюватиметься особливий файл (файловий дескриптор) зі шляхом, який дасть змогу повідомляти всі локальні процеси за допомогою читання або запису з цього файлу. Сокети INET типу можуть бути доступні з мережі та потребують виділення для нього унікального номера порту[2]. У більшості випадків клієнт під'єднується до слухача, після чого запис або читання через файловий дескриптор сокету будуть обмінювати інформацію між сервером і ним. Сокет зазвичай, на практиці, буде посилатися на сокет в Internet Protocol мережі (де сокет буде мати називатися інтернет-сокетом), як приклад для протоколу з'єднань один до одного, тобто Transmission Control Protocol (TCP). У цьому випадку очікується, що сокети будуть прив'язані до певної адреси сокету, а саме IP-адреси і номеру порту, що використовується для локального вузла, і відповідної адреси сокету. Прив'язка сокета з адресою іншого сокету носить

назву "binding", що переводиться як прив'язування.[1,3].Маємо на увазі те, що в той момент, коли локальний процес має змогу спілкування з іншим процесом, відправляючи та отримуючи інформацію з зовнішньої сокет-адреси, він не матиме доступу до неприв'язаного сокета, і не має змоги використання неприв'язаний сокет. Доступу до файлового дескриптор сокета нема, так як вони являються внутрішніми для зовнішнього вузла. Приклад цієї ситуації, при зв'язку між 11.22.33.44:5555 і 55.66.77.88:9999 (локальною адресою: локальним портом, зовнішньою адресою: зовнішнім портом) буде створений зв'язаний сокет на кожному з кінців, який відповідає внутрішньому з'єднанню зі стеком деяких протоколів на вузлі. Вони мають назву дескриптори числових сокетів, скажімо 111 на одній стороні і 222 на іншій. Процес на вузлі 11.22.33.44 запрошує зв'язок із вузлом 55.66.77.88 на порту 9999 (це запит, для створення стеком протоколів сокета для спілкування з адресою 55.66.77.88), і як тільки створюється сокет і отримує файловий дескриптор сокета 111, він може спілкуватися за допомогою цього сокета через файловий дескриптор 111. Стеком протоколу буде перевідправляти інформацію на вузол 55.66.77.88 і з нього на порт 9999. Але процес який знаходиться на вузлі 11.22.33.44 не зможе мати зв'язок на основі файлового дескриптора зовнішнього сокета, так як вони являються внутрішніми для другого вузла і не мають змоги бути використаним стеком протоколів на вузлі 11.22.33.44[3]

3. Датчики температури

3.1. Датчик температури та вологості DHT11

DHT11 - це цифровий датчик вологості і температури, що складається з термістора і ємнісного датчика вологості. Також датчику знаходиться аналого-цифровий перетворювач для перетворення значень вологості та температури. Датчик DHT11 не володіє високою швидкістю і точністю, але зате простий, недорогий і відмінно підходять для навчання і контролю вологості та температури в приміщенні.

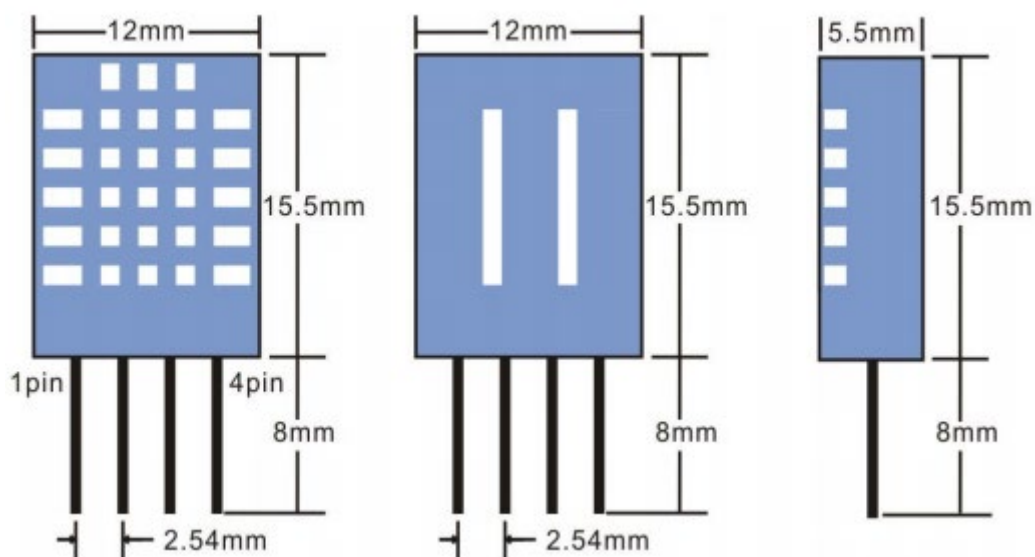


Рис. 3.1.1 Розміри датчику DHT11 у трьох проекціях

Таблиця 3.1.2. Характеристики датчику DHT11

Живлення	3,5 - 5,5 В
Чутливий елемент	Полімерний резистор
Струм живлення в режимі вимірювання	0,3 мА
Струм живлення в режимі очікування	60 мкА

Температурна чутливість	0,1 ° C
Роздільна здатність виміру вологості	1%
Діапазон виміру вологості	20%-80%
Діапазон виміру температури	0-50 ° C
Похибка вимірювання вологості	+5%
Похибка вимірювання температури	+1 ° C
Частота оновлення показників	1 Гц

Розміри сенсору зазначені на рис.3.1.1.

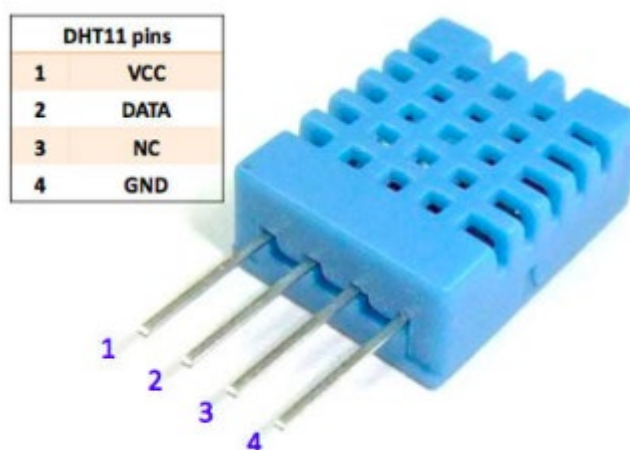


Рис.3.1.3. Розпіновка сенсору температури DHT11

Сенсор температури має чотири піни, один з яких не використовується, а саме номер 3. На рис.3.1.3. зазначена розпіновка DHT11, згідно якої:

1. Живлення
2. Пін для передачі даних
3. Не використовується
4. Земля

3.2. Датчик температури та вологості DHT22

Датчик вологості і температури DHT22 (кодова назва AM2303) - це два датчика в одному корпусі, результати вимірювання яких передаються на цифровий блок з аналого-цифровим перетворювачем (для датчика відносної вологості) і на виході з датчика виходить цифровий сигнал (контакт DATA). Для роботи датчика на контакт VCC подається напруга 3.3-6В. DHT22 має дуже низьке енергоспоживання. Датчик відкалібрований на заводі.

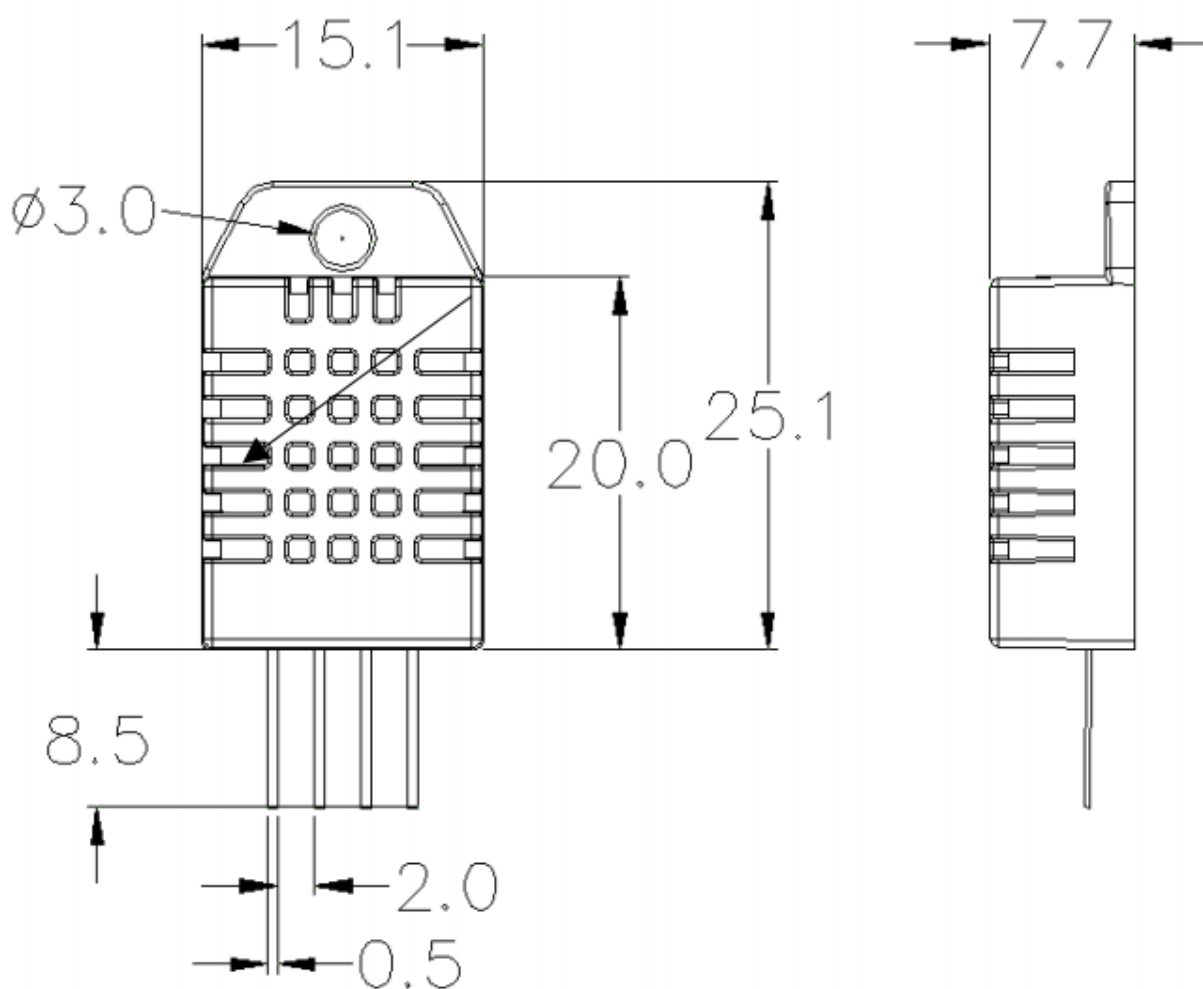


Рис.3.2.1. Розміри датчику DHT22 у двох проєкціях

На рис.3.2.1. зазначен зовнішній вигляд та розміри датчику. Як видно з нього, він має дещо більші розміри ніж датчик DHT11.

Таблиця 3.2.2. Характеристики датчику DHT22

Живлення	3,5 - 6 В
Чутливий елемент	Полімерний конденсатор
Струм живлення в режимі вимірювання	0,3 мА
Струм живлення в режимі очікування	60 мкА
Температурна чутливість	0,1 ° С
Роздільна здатність виміру вологості	1%
Діапазон виміру вологості	0%-100%
Діапазон виміру температури	-40-80 ° С
Похибка вимірювання вологості	+2%
Похибка вимірювання температури	+0,5 ° С
Частота оновлення показників	0,5 Гц

В таблиці 3.2.2 зазначені характеристики датчику DHT22, зпираючись на які ми будемо вибирати датчик температури.

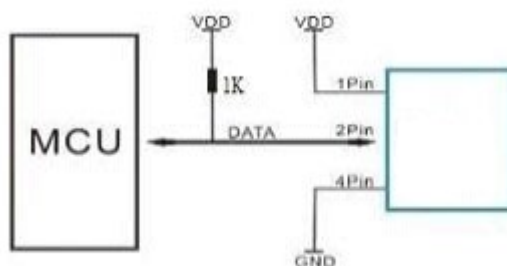


Рис.3.2.3. Розпіновка датчику DHT22

Сенсор температури має чотири піни, один з яких не використовується, а саме номер 3. На рис.3.2.3. зазначена розпіновка DHT22, згідно якої:

1. Живлення
2. Пін для передачі даних
4. Земля

3.3. Вибір робочого датчик

Обидва датчики, сімейства DHT влаштовують тим, що вони одночасно можуть міряти як температуру, так й вологість повітря. Для роботи з цими датчиками, є готовий код, що декодує інформацію, яка поступає до клієнту.

Зробимо порівняльну таблицю цих двох датчиків

Таблиця 3.3.1 Порівняння характеристик DHT11 та DHT22

	DHT11	DHT22
Живлення	3,5 - 5,5 В	3,5 - 6 В
Чутливий елемент	Полімерний резистор	Полімерний конденсатор
Струм живлення в режимі вимірювання	0,3 мА	0,3 мА
Струм живлення в режимі очікування	60 мкА	60 мкА
Температурна чутливість	0,1 ° С	0,1 ° С
Роздільна здатність виміру вологості	1%	1%
Діапазон виміру вологості	20%-80%	0%-100%

Діапазон виміру температури	0-50 ° C	-40-80 ° C
Похибка вимірювання вологості	+5%	+2%
Похибка вимірювання температури	+1 ° C	+0,5 ° C
Частота оновлення показників	1 Гц	0,5 Гц

Як видно з таблиці 3.3.1, датчик DHT22, у порівняння з датчиком DHT11, має такі переваги:

- розширений температурний діапазон
- розширений діапазон виміру вологості
- меншу похибку виміру вологості
- меншу похибку виміру температури

Але й такі недоліки:

- швидкодія менша у 2 рази
- має вищу ціну, приблизно у 5 разів.

Виходячи за таких переваг та недоліків, я обираю DHT22 в якості робочого сенсору, тому що:

- у DHT11 суттєва похибка при вимірюванні температури та вологості + малий діапазон вимірювання.
- незважаючи на те, що ціна DHT22 у 5 разів вища, ніж у DHT11, відносно вартості Raspberry Pi 3B, це не буде суттєвою сумою.

4. Штучні нейронні мережі

4.1 Історія розвитку штучних нейронних мереж

Вважається, що теорія нейронних мереж як наукова дисципліна вперше була позначена в класичній праці Уорреном Мак-Коллока і Уолтера Питтса в 1943 р, в якій стверджувалося, що практично будь-яку логічну операцію можна реалізувати за допомогою простої нейронної мережі. Продовжуючи дослідження в цій області, в 1958 р Френк Розенблат винаходить одношаровий перцептрон, який є однією з перших моделей нейронної мережі і демонструє його здатність вирішувати завдання класифікації. Однак, як виявилось пізніше, перцептрон мав серйозні обмеження. У 1969 році Марвін Лі Мінський опубліковує в статті формальний доказ про нездатність перцептрону вирішувати деякі завдання, написавши про це цілу книгу. Лише через 20 років, нейронні мережі стали активно використовуватися, що пов'язано з запропонованим енергетичним підходом Джона Хопфілда та створенням алгоритму зворотного поширення помилки, який вперше був запропонований Вербосом та незалежно був розроблений рядом інших авторів. Алгоритм отримує популярність завдяки Румельхарту в 1986 р. Цей алгоритм активно використовується в наш час.

З середини 80-х років теорія нейронних мереж отримала ще більший імпульс дослідження в зв'язку з появою потужних персональних комп'ютерів [1].

4.2. Задачі, котрі вирішують штучні нейронні мережі

На сьогоднішній момент, коло завдань, які можуть бути вирішені з використанням нейронних мереж досить великий. До найпоширеніших можна віднести:

- Розпізнавання образів і класифікація. Суть полягає в розпізнаванні вхідного образу (символів тексту, зображення, мовних сигналів і т.д.) і

надання його належності до певного класу. Під час навчання нейронної мережі, на вхід їй подають вектор значень ознак образу, з зазначенням його класу. Після навчання, їй можна подавати невідомі раніше образи і отримувати відповідь про належність до певного класу. У такого роду завданнях, встановлюється відповідність між вихідним шаром штучної нейронної мережі і класом, який він представляє.

- Завдання кластеризації. Завдання кластеризації, також відомі як класифікація образів "без вчителя", не мають навчальної вибірки з мітками класів. Їх принцип роботи заснований на виявленні закономірностей і стереотипів між образами і розміщенні цих образів в один кластер або категорію [3].
- Прогнозування. Здатність нейронних мереж до передбачення результатів обумовлена узагальненням і виділенням загальних залежностей між вхідними та вихідними даними. Після навчання, мережа здатна передбачити майбутнє значення деякої послідовності, ґрунтуючись на її попередні зміни. Для того щоб штучна нейронна мережа могла дати максимально точну відповідь, необхідно навчати її на даних, в яких дійсно простежується якась залежність. В іншому випадку, прогнозування швидше за все не дасть ніяких результатів
- Апроксимація функцій. Завдання апроксимації, полягає в знаходженні оцінки невідомої функції, по її значенням. Точність апроксимації залежить від вибору структури нейронної мережі [3].
- Оптимізація. Під задачею оптимізації з використанням нейронних мереж, мають на увазі знаходження такого рішення, яке задовольняє системі умов і мінімізує цільову функцію. Як приклад можна розглянути алгоритм, який реалізує якусь послідовність дій, який можна повністю замінити функціонуванням нейронної мережі, використовуючи правила, за якими він складений [3].

4.3. Штучний нейрон

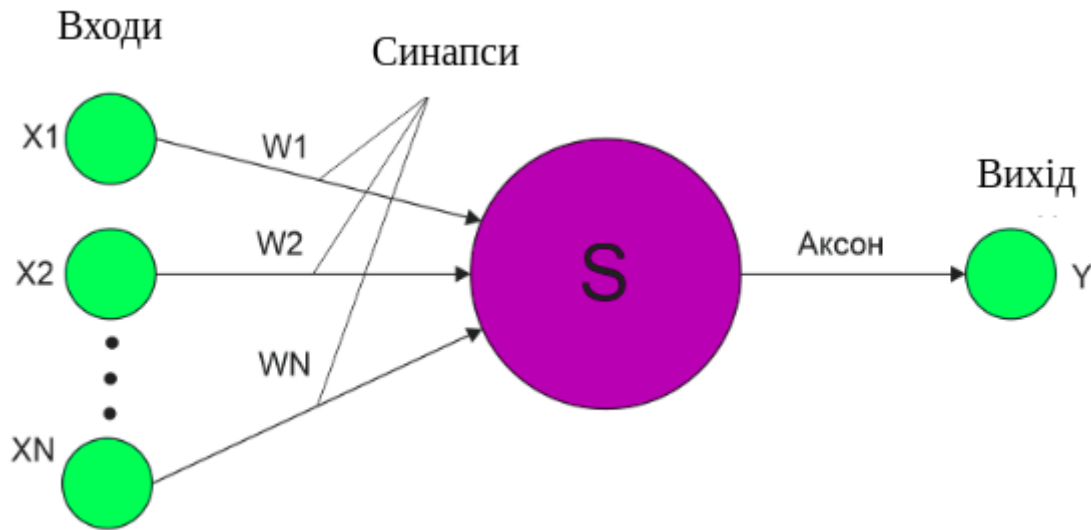


Рис.4.3.1. Модель штучного нейрону

Штучний нейрон є елементарною структурною одиницею штучної нейронної мережі і являє собою спрощену модель біологічного нейрона. На рис.4.3.1 зображено нейрон, входами якого можуть бути або вхідні дані, або виходи від такого ж нейрона. Входи з'єднані з осередком нейрона S за допомогою синаптичних зв'язків. Кожен синапс має свою вагу, при передачі в осередок нейрона вхідного параметра, він відповідно множиться на вагу, тобто $x_i \cdot w_i$.

$$I = \sum_{i=1}^n x_i \cdot w_i$$

(4.3.1)

Стан нейрона S , визначається за формулою 4.3.1.

$$S = f(I)$$

(4.3.2)

Вихід нейрона визначається формулою 4.3.2.

Функція f називається функцією активації. Такі функції не можуть бути лінійними, оскільки нейронні мережі з лінійною функцією активації ефективні

тільки на одному рівні, незалежно від того, наскільки складна їх структура. Однією з найбільш поширених функцій є нелінійна функція з насиченням, так звана логістична функція або сигмоїда:

$$f(x) = \frac{1}{1 + e^{-ax}} \quad (4.3.3)$$

Параметр a відповідає за пологість функції. Чим він менше, тим більш пологою стає сигмоїда. Слід зазначити, що дана функція диференційована на всій осі абсцис, що є необхідною умовою до функцій активації.

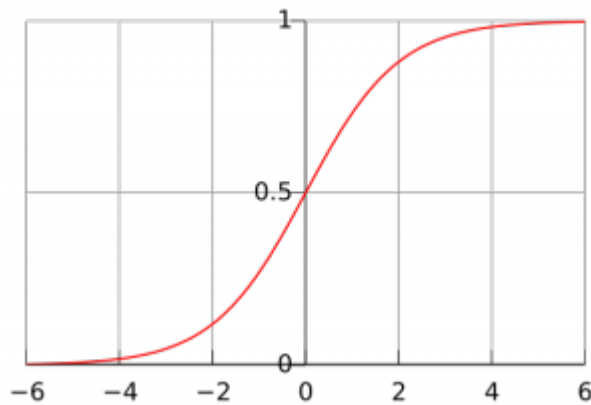


Рис.4.3.2. Сигмоїда

Крім того, вона має властивість підсилювати слабкі сигнали краще, ніж великі, а також запобігає насиченню від великих сигналів так як вони відповідають областям аргументів, де сигмоїда має пологий нахил, що видно з рис. 4.3.2[2].

4.4. Архітектури штучних нейронних мереж

Можна виділити кілька основних типів нейронних мереж:

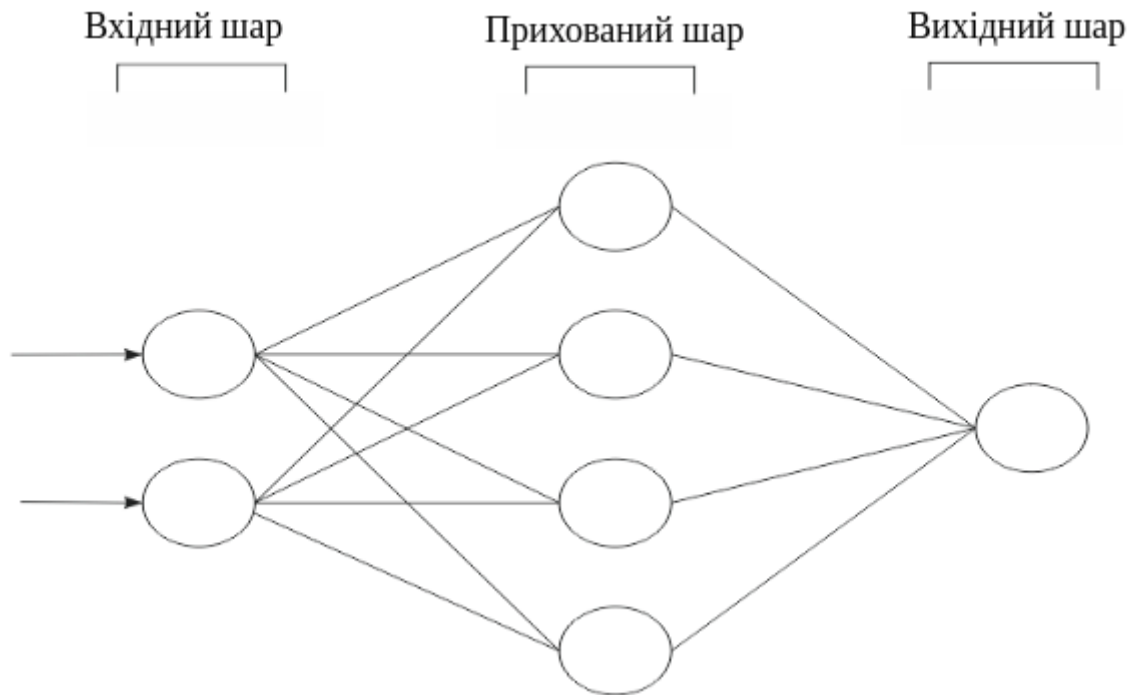


Рис.4.4.1. Багатошарова штучна нейронна мережа

Багатошарові мережі. У багатошарових мережах один або кілька нейронів об'єднуються в шари рис.4.4.1. Шар - сукупність нейронів, на вхід яких подається один і той же загальний сигнал. У мережах такого типу, зовнішні вхідні дані подаються на входи нейронів першого шару, а вихідні дані є результатом останнього вихідного шару. Крім вхідного і вихідного шарів, в багатошарових мережах так само присутній один або кілька прихованих шарів. Зв'язки виходів нейронів від деякого шару i до деякого шару $i + 1$ називають послідовним.

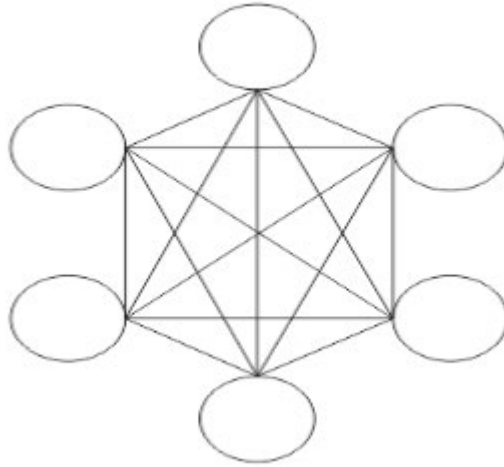


Рис.4.4.2. Повнозв'язна штучна нейронна мережа

Повнозв'язні нейронні мережі. У повнозв'язної нейронної мережі кожен нейрон передає свій сигнал іншим нейронам рис.4.4.2.

Вихідними сигналами, можуть бути всі або деякі сигнали нейронів після кількох тактів функціонування. Всі вхідні сигнали, подаються на вхід всіх нейронах [6]. Кожна нейронна мережа має не тільки свою архітектуру, а й тип, який краще підходить для вирішення конкретного завдання. Наприклад, згорткові нейронні мережі (convolutional neural network CNN) набагато краще справляється з розпізнаванням образів і задачами комп'ютерного зору.

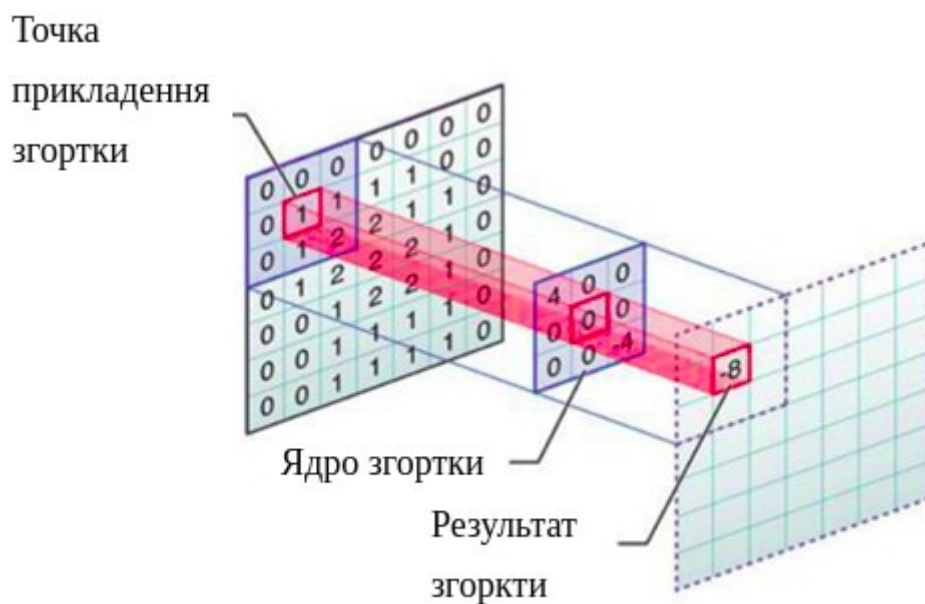


Рис.4.4.3. Згорткова штучна нейронна мережа

Її відмінність від інших типів штучної нейронної мережі полягає в тому, що кожен фрагмент зображення множиться на матрицю (ядро) згортки поелементно, а результат підсумовується і записується в аналогічну позицію вихідного зображення рис.4.4.3. Якщо кожен піксель зображення розглядалося б окремо, це призвело б мережу до швидкого перенавчання та її здатність розпізнавання образів була б точна тільки на навчальній вибірці [7].

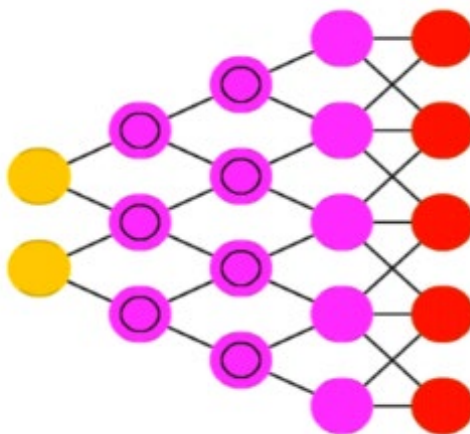


Рис.4.4.4. Розгорткова штучна нейронна мережа

Існують так само розгорткові нейронні мережі (deconvolutional networks DN), які ще називають зворотними графічними мережами і є зворотними до згорткових нейронних мереж рис.4.4.4. Їх завдання генерувати зображення по заданих ознаках. Наприклад, при передачі мережі слова "кіт" вона повинна буде згенерувати зображення схожі на котів.

Рекурентні нейронні мережі. Рекурентними називають штучні нейронні мережі, в яких, поряд із прямими зв'язками, направленими від входів (рецепторів) мережі до її виходів (ефекторів), є зворотні, що мають протилежний напрямок. На відміну від штучних нейромереж прямого поширення, які здійснюють статичну проекцію поданих на входи векторів даних (стимулів) у вихідні

вектори (реакції), рекурентні нейронні мережі є динамічними системами, що оперують з послідовностями вхідних даних, перетворюючи їх на послідовності реакцій. Поведінка рекурентних нейромереж віддзеркалює набуті при навчанні стереотипи, що робить їх близькими до цілеспрямованих адаптивних динамічних систем, запрограмованих на досягнення заздалегідь визначених цілей. Але на відміну від останніх програмування поведінки рекурентних нейромереж здійснюється шляхом навчання на прикладах, що не потребує формального визначення цілей. Вони здатні ефективно діяти в умовах невизначеності, зокрема, вирішувати задачі адаптивного керування поведінкою складних систем у нестационарному оточенні, приймати оперативні рішення в системах ситуаційного управління тощо.

Термін “рекурентна нейронна мережа” набув поширення наприкінці 80-х років, коли було запропоновано методи навчання нейромереж із зворотними зв'язками та визначено основні проблеми, що вимагали вирішення на шляху до їх реалізації та практичного застосування [4–6]. Найбільші складності викликали висока ресурсоемність та незадовільна результативність процесу навчання, а також відсутність гарантії стабільності поведінки рекурентних нейромереж. За минулі два десятиріччя досягнуто суттєвого прогресу у вирішенні цих проблем. Розроблено більш ефективні методи навчання та тестування рекурентних нейромереж, розвинуто загальну теорію динамічних рекурентних нейромереж, запропоновано нові архітектурні рішення, які дозволяють поєднувати рекурентні структури з відомими нейропарадигмами і одержувати нейромережі з новими властивостями [7].

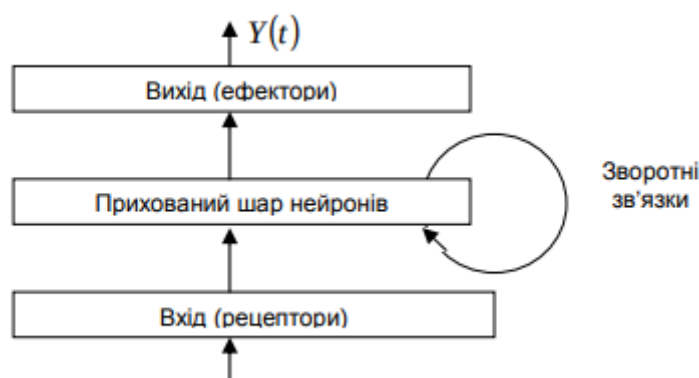


Рис.4.4.5 Загальна структура рекурентної нейронної мережі

Будову рекурентної нейронної мережі схематично зображено на рис. 4.4.5. [8]. Її входам та виходам відповідають шари рецепторних та ефекторних нейронів або просто лінії передачі даних. Між цими шарами розташовано один або кілька шарів прихованих нейронів. Входи нейронів кожного шару мають прямі зв'язки з виходами нейронів попереднього шару та можуть мати зворотні зв'язки з виходами нейронів свого та наступних шарів.

4.5. Довга короткочасна пам'ять

Довга короткочасна пам'ять - це архітектура РНМ, запропонована 1997 року Зеппом Хохрайтером та Юргеном Шмідгубером. На відміну від традиційних рекурентних нейронних мереж, мережа довгої короткочасної пам'яті прекрасно підходить для процесу навчання в задачах з метою класифікації, прогнозування у часових рядів, сигналах, коли між важливими значеннями є часові затримки з невідомою довжиною. Не чутливість до довжини невідомих прогалин дає ячeyці довгої короткочасної пам'яті перевагу в багатьох застосуваннях над іншими рекурентними нейронними мережами, такі як ванільна рекурентна мережа та GRU, прихованими марковськими моделями та іншими математичними моделями для навчання на послідовностях. Інші успіхи довгої короткочасної пам'яті в досягненні найкращих результатів у кодуванні тексту природною мовою,[15] у задачі розпізнаванні несегментованого рукописного тексту, та у 2009 році посіла перше місце у змаганні з розпізнавання тексту написаного від руки ICDAR. Мережі довгої короткочасної пам'яті також застосовуються у автоматичному розпізнаванні голосу, та були головною часткою мережі, яка 2003 року добралась рекордного рівня у 17.7% фонемних помилок на класичному академічному датасеті природного мовлення TIMIT.

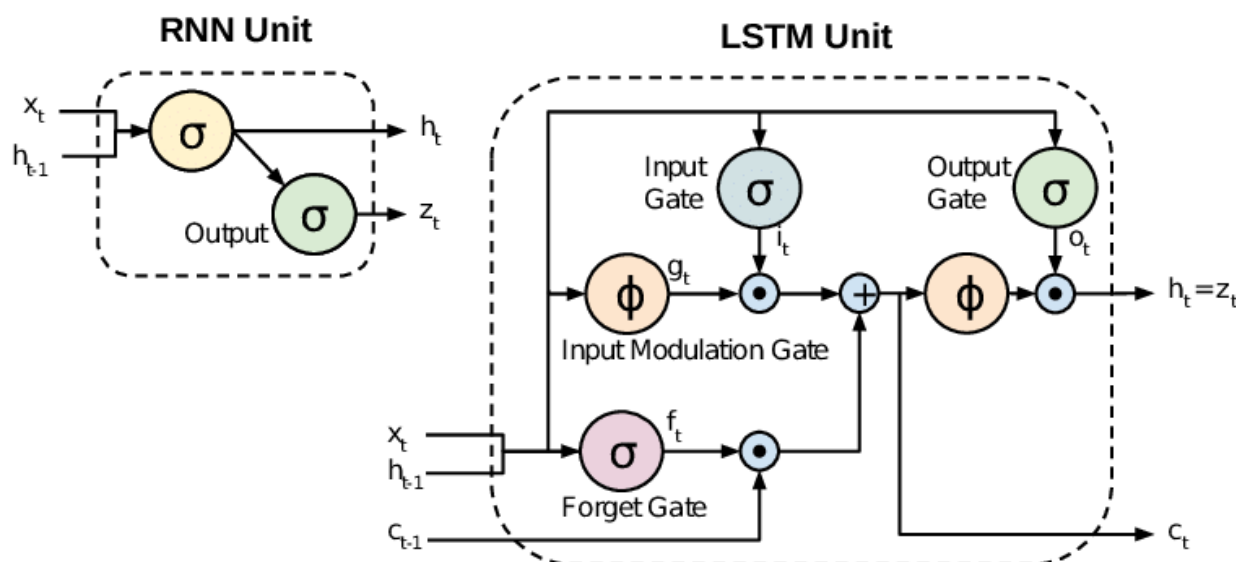


Рис. 4.5.1. Порівняння “ванільної” комірки рекурентної нейронної мережі та комірки довгої короточасної пам’яті.

Блоки ДКЧП містять три або чотири «вентилі» (англ. gates) рис.4.5.1, які вони використовують для керування взаємодією з інформацією до або з їх пам'яті. Ці вентилі реалізуються з використанням логістичної функції для отримання значень між 0 та 1. Для часткового пропуску або закриття ходу інформації з або до цієї пам'яті використовується множення на це значення. Тобто, «вхідний вентиль» (input gate) керує мірою, що визначає як нове значення потрапляє до пам'яті. Forget gate, тобто забувальний вентиль, керує мірою, яка визначає значення наскільки залишається в пам'яті. А output gate, тобто вихідний вентиль, керує мірою, до якого величина в пам'яті обчислюється з активації вихідного блоку. (Іноді зустрічається ячейки, в яких входовий та забувальний елементи з'єднують в один. Ідея цього з'єднання мотивується тим, що час забування настає після отримання нового значення, що варте, щоб його запам'ятали.)

4.6. Навчання штучної нейронної мережі

Процес навчання нейронної мережі є необхідним, для її здатності виконувати поставлені завдання. Цей процес може бути розглянутий як налаштування архітектури мережі і ваг зв'язків за допомогою моделювання середовища, в яку ця мережа вбудована. Властивість нейронної мережі навчатися на прикладах дозволяє спростити завдання умов для вирішення конкретного завдання, в порівнянні з системами, які слідує набору певних правил, складених експертами. Розділяють алгоритми навчання з вчителем і без вчителя [3].

Процес навчання з учителем передбачає існування вибірки навчальних прикладів. Кожен зразок подається на вхід нейронної мережі, потім проходить процедуру обробки всередині штучної нейронної мережі. Після обчислення вихідного сигналу штучної нейронної мережі порівнюється отриманий результат з відповідним значенням цільового вектора, що представляє собою необхідний вихід мережі. Обчисливши помилку, відбувається зміна вагових коефіцієнтів зв'язків всередині мережі за обраним алгоритмом. Ваги підлаштовуються під кожен вектор до тих пір, поки помилка по всьому масиву вхідних даних не досягне заданого рівня.

Навчання без вчителя не вимагає знання правильної відповіді на кожен приклад навчальної вибірки. В цьому випадку розкривається внутрішня структура даних або кореляція в системі даних, що дозволяє розподілити образи за категоріями.

Залежно від розв'язуваної задачі в навчальній вибірці використовуються ті чи інші типи даних і різні розмірності вхідних / вихідних сигналів. Як приклад вхідних даних навчальної вибірки - зображення, таблиці чисел, розподілення. Типи вхідних даних - бінарні (0 і 1), біполярні (-1 і 1) числа, цілі або дійсні числа з деякого діапазону. Вихідні сигнали мережі - вектора цілих або дійсних чисел. Для вирішення практичних завдань часто потрібні навчальні вибірки великого обсягу. Через жорстко обмеженого обсягу оперативної пам'яті комп'ютера розмістити в ній великі навчальні вибірки неможливо. Тому вибірка ділиться на пакети - групи прикладів. У кожен момент часу лише один пакет прикладів розташовується в пам'яті комп'ютера, інші - на жорсткому диску.

Сторінки послідовно завантажуються в пам'ять комп'ютера. Навчання мережі відбувається по всій сукупності пакетів прикладів, по всій навчальній вибірці.

В даний час відсутня універсальна методика побудови навчальних вибірок. Набір навчальних прикладів формується за бажанням інженера машинного навчання, що навчає конкретну нейронну мережу, індивідуально для кожної конкретної розв'язуваної задачі [4].

Якщо в ненавчену нейронну мережу ввести вхідний сигнал одного з прикладів навчальної вибірки, то вихідний сигнал мережі буде істотно відрізнятися від бажаного вихідного сигналу, визначеного в навчальній вибірці. Функція помилки чисельно визначає подібність всіх поточних вихідних сигналів мережі і відповідних бажаних вихідних сигналів навчальної вибірки. Найбільш поширеною функцією помилки є середньоквадратичне відхилення. Однак запропоновані й інші функції помилки.

Для навчання нейронних мереж можуть бути використані різні алгоритми. Можна виділити дві великі групи алгоритмів - градієнтні і стохастичні. Градієнтні алгоритми навчання мереж засновані на обчисленні часткових похідних функції помилки від параметрів мережі. Серед градієнтних розрізняють алгоритми першого і другого порядків. В стохастичних алгоритмах пошук мінімуму функції помилки ведеться випадковим чином[2].

При навчанні мереж, як правило, використовується один з двох наступних критеріїв зупинки навчання: зупинка при досягненні деякого мінімального значення функції помилки, зупина в разі успішного вирішення всіх прикладів навчальної вибірки.

Перед навчанням виконується ініціалізація нейронної мережі, тобто присвоювання параметрам мережі деяких початкових значень. Як правило, ці початкові значення - деякі малі випадкові числа.

Для формування навчальних вибірок, ініціалізації і навчання в програмах моделювання нейронних мереж використовуються спеціальні процедури. Можливість використання пакетного навчання є дуже важливою при вирішенні

практичних завдань за допомогою нейронних мереж, що моделюються на звичайних комп'ютерах.

Навчання - це ітераційна процедура, яка при реалізації на звичайних комп'ютерах, вимагає значного часу. Алгоритми навчання істотно розрізняються по швидкості збіжності. Однією з найважливіших характеристик програм для моделювання нейронних мереж є швидкість збіжності алгоритму (або алгоритмів) навчання, які реалізовані в програмі[2].

Теорія навчання розглядає три фундаментальні властивості, пов'язаних з навчанням за прикладами: ємність, складність зразків і обчислювальна складність. Під ємністю розуміється, скільки зразків може запам'ятати мережа, і які функції і межі прийняття рішень можуть бути на ній сформовані. Складність зразків визначає число навчальних прикладів, необхідних для досягнення здатності мережі до узагальнення. Занадто мала кількість прикладів може викликати "перенавчання" мережі, коли вона добре працює на прикладах навчальної вибірки, але погано - на тестових прикладах, підлеглих того ж статистичному розподілу. Відомі 3 основних типи правил навчання: корекція помилково, машина Больцмана і правило Хебба.

Правило корекції помилки. При навчанні з учителем для кожного вхідного прикладу заданий бажаний вихід d . Реальний вихід мережі u може не збігатися з бажаним. Принцип корекції помилки при навчанні полягає у використанні сигналу $(d-u)$ для модифікації ваг, що забезпечує поступове зменшення помилки. Навчання має місце тільки в разі, коли персептрон помиляється. Відомі різні модифікації цього алгоритму навчання [2].

Навчання Больцмана. Являє собою стохастичне правило навчання, яке впливає з інформаційних теоретичних і термодинамічних принципів. Метою навчання Больцмана є така настройка вагових коефіцієнтів, при якій стани видимих нейронів задовольняють бажаного розподілу ймовірностей. Навчання Больцмана може розглядатися як спеціальний випадок корекції помилки, в якому під помилкою розуміється розбіжність кореляцій станів в двох режимах.

Правило Хебба. Найстарішим навчальним правилом є постулат навчання Хебба. Хебб спирався на наступні нейрофізіологічні спостереження: якщо нейрони з обох сторін синапсу активізуються одночасно і регулярно, то сила синаптичного зв'язку зростає. Важливою особливістю цього правила є те, що зміна синаптичної ваги залежить тільки від активності нейронів, які пов'язані з даними синапсом. Це істотно спрощує ланцюг навчання.

Для того щоб перевірити навички, набуті нейронною мережею в процесі навчання, використовується імітація функціонування мережі. У мережу вводиться деякий сигнал, який, як правило, не збігається ні з одним з вхідних сигналів прикладів навчальної вибірки. Далі аналізується отриманий вихідний сигнал мережі. Тестування навченої мережі може проводитися на одиночних вхідних сигналах, або на контрольній вибірці, яка має структуру, аналогічну навчальній вибірці [2].

4.7. Алгоритм зворотного поширення помилки

Алгоритм зворотного поширення помилки (back propagation) відноситься до методу навчання з корекцією помилки і, як правило, застосовується до багатошарових нейронних мереж. Це метод навчання "з вчителем", при якому "наставник" навчає мережу, також як дитину навчають читати і писати. При навчанні на вхідний шар багаторазово подаються образи сигналів, розпізнаванню яких нейронна мережа повинна бути навчена, і коригуються ваги нейронів для досягнення бажаного вихідного сигналу. Для поліпшення якості розпізнавання образи, що подаються на вхідний шар, можуть бути злегка змінені (добавлений шум і т.п.). Детально процедура навчання виглядає так:

- 1) Вибірка вхідних даних (безліч образів, класифікованих учителем) розбивається на дві: навчальну і контрольну послідовності. Зазвичай навчальна послідовність містить більше образів, ніж перевірочна.

- 2) Проводиться ініціалізація всіх ваг, включаючи граничні, невеликими випадковими величинами (зазвичай в діапазоні $[-1; +1]$). Це визначає початкову точку на поверхні функції помилки для методу градієнтів, позиція може виявитися вирішальною для збіжності мережі.
- 3) Проводиться прямий прохід мережі для першого способу з навчальної вибірки від вхідного шару через приховані шари до вихідного шару: кожен нейрон підсумовує результат множення входів на ваги і видає результат функції активації, застосованої до цієї суми, на нейрони наступного рівня.
- 4) Обчислюється різниця між дійсним і бажаним вихідним значенням кожного нейрона вихідного шару. При їх розбіжності має місце помилка в розпізнаванні (класифікації) образів.
- 5) Проводиться процедура зворотного поширення цих помилок зі зв'язків від вихідних нейронів до вхідних і визначаються помилки для кожного нейрона. Розглянемо цю процедуру докладніше. Припустимо, вихідне значення нейронної мережі було 0.5, а бажане 0.

$$E = d_i - a_i \quad (4.7.1)$$

Нехай помилка визначається за формулою 4.6.1. Де E - помилка мережі, y_i - бажане значення на i -му виході нейронної мережі, d_i - вихідне значення мережі на i -му виході.

В даному випадку, на виході помилка буде дорівнює $E = 0.5 - 0 = 0.5$.

$$\Delta w = w * \text{sigmoid}'(x) \quad (4.7.2)$$

Якщо розглядати в якості функції активації сигмоид, то розрахунок Δw , на який необхідно зрушити ваги синаптичних зв'язків передостаннього останнього шару, буде здійснюватися за формулою 4.6.2.

Тут $\text{sigmoid}(x) dx$ - похідна функції активації, рівна $\text{sigmoid}(x) (1 - \text{sigmoid}(x))$. Під x мається на увазі значення, яке було отримано шляхом підсумовування всіх

вихідних значень нейронів передостаннього шару, відповідно помножених на w_i .

$$\square_{\square} = \square_{\square} - \square_{\square} * \square_{\square} * \square \quad (4.7.3)$$

Завершальний крок корекції ваг синаптичних зв'язків між останнім і передостаннім шаром буде проводитися за формулою 4.6.3, m_i - вихідні значення і нейрона передостаннього шару, l - параметр відповідає за швидкість навчання.

Після того як ваги порашовані, ми можемо порашувати помилку вихідного значення m_i за формулою $E = w_i * \Delta w$. Знаючи помилку кожного нейрона на попередньому шарі, застосовуючи формули, які використовувалися вище, можемо знайти відхилення ваг і значення нейронів на попередніх шарах.

б) Знову проводиться прямий прохід мережі вже для наступного образу навчальної вибірки.

Кроки 3-6 повторюються до досягнення деякого критерію, наприклад, досягнення помилкою встановленої межі.

По завершенню навчальної фази мережу перевіряється за допомогою контролюючої послідовності, що містить образи, не пред'явлені раніше. (При цьому не проводиться корекція ваг, а лише обчислюється помилка). У разі, якщо якість роботи знайдено задовільним, мережа вважається готової до роботи. В іншому випадку мережа піддається повторному навчання, при якому можлива зміна деяких параметрів (початкові ваги, кількість нейронів в прихованих шарах, додаткові навчальні образи і т.п.) [2].

4.8 Проблеми штучних нейронних мереж

Однією з головних проблем нейронних мереж є необхідність в дуже великих обсягах даних. Кілька років тому, були продемонстровані алгоритми, здатні

розпізнавати образи на зображеннях краще, ніж людина. Що б перевершити людину, мереж довелося вивчити більше 1,2 мільйона зображень, в той час як дитина, може навчитися визначати новий об'єкт або тварину після однієї побаченої фотографії. Практично для будь-якої задачі розпізнавання образів, штучних нейронних мереж необхідно в сотні тисяч разів більше інформації ніж людині.

Іншою проблемою, є не пристосованість нейронних мереж до мультизадачності. Сучасні алгоритми призначені тільки для вирішення одного завдання. ШНМ можна навчити розпізнавати собак або створювати музику. Але на сьогоднішній момент, не існує таких мереж, які могли б виконувати обидві ці задачі. Якщо розглянути нейронну мережу з фіксованою кількістю шарів і нейронів в кожному шарі, то при постійному збільшенні, наприклад, навчальних образів, рано чи пізно нейронна мережа перестане піддаватися обробці. Або якщо вже навчену нейронну мережу розпізнавати обличчя, почати навчати розпізнавати кішок, то вона почне забувати про особи для звільнення пам'яті для нової інформації [5].

У 2016 році був проведений експеримент. Дослідники хотіли визначити, на яку частину зображення "дивиться" нейронна мережа щоб виконати завдання. Вони показали нейронної мережі фотографію спальні і запитали у неї "Що висить на вікнах?". Замість того, щоб подивитися на вікна, ІНС стала дивитися на підлогу, після чого на ліжку, з чого зробила висновок, що на вікні висять фіранки. Справа в тому, що неможливо подивитися "вглиб" нейронних мереж, для того щоб зрозуміти, як вони працюють.

Так само, однією з проблем нейронних мереж є її правильне проектування, наприклад, невідомо скільки шарів необхідні для даного завдання, скільки потрібно вибрати елементів у кожному шарі, як мережа буде реагувати на дані, які не включені в навчальну вибірку (здатність мережі до узагальнення) і який розмір навчальної вибірки необхідний для здатності мережі до узагальнення. Найчастіше, дані параметри визначаються шляхом проб і помилок. Існуючі

теоретичні результати дають лише слабке уявлення про те, якими мають бути ці параметри [5].

5. Навчання штучної нейронної мережі

5.1. Архітектура штучної нейронної мережі

В якості штучної нейронної мережі я обрав рекурентну нейронну мережу з довгою короткочасною пам'яттю. Вибрана ця архітектура не випадково, так як я аналізуючи часовий ряд, де основна величина температура та вологість, ми не можемо гарантовано знати, наскільки довга часова залежність а також є часова залежність, а, як згадувалось вище, рекурентна нейронна мережа з довгою короткочасною пам'яттю в змозі сама вивчити і зрозуміти цю залежність і нам не треба сильно піклуватися за довжину часового ряду, який ми будемо використовувати при навчанні та прогнозуванні. Також використання ячейки з довгою короткочасною пам'яттю дозволить уникнути нам таких неприємних явищ, як вибух градієнтів, та зникнення градієнтів. З плюсів використання ячейки з довгою короткочасною пам'яттю дозволить нейронній мережі самій зрозуміти наскільки довга існує залежність у часі.

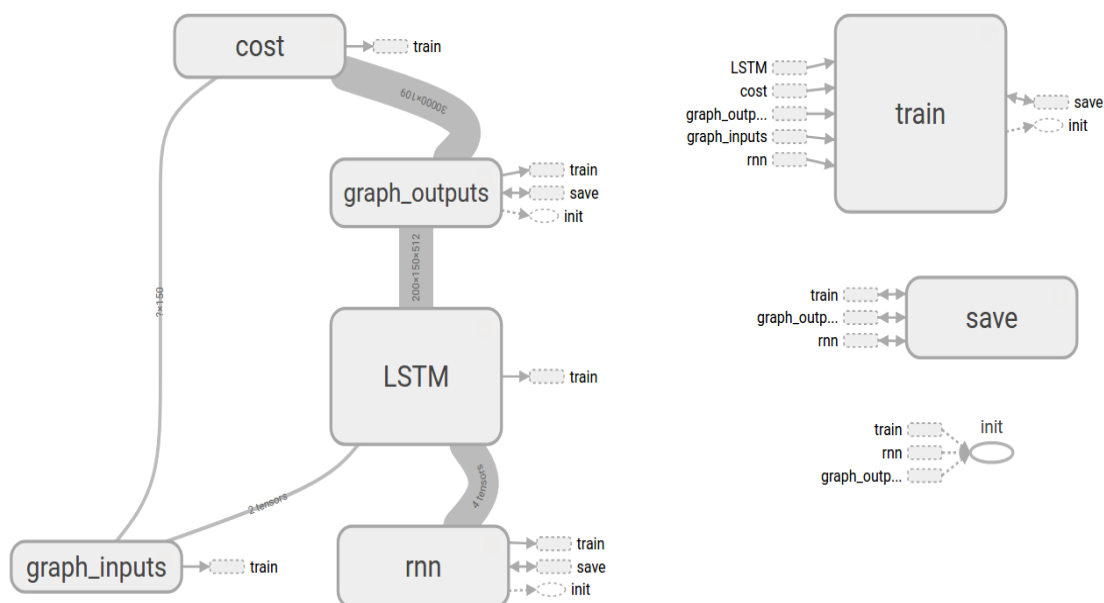


Рис. 5.1.1 Обчислювальний граф штучної нейронної мережі

На рис. 5.1.1. показаний обчислювальний граф, вигляд якого був отриманий за допомогою додатку TensorBoard, що йде у комплекті з бібліотекою глибоко навчання Tensorflow, що була розроблена компанією Google[17]. Нею я й буду користуватися в подальшому.

5.2 Дані для навчання

Дані для навчання були зібрані мною у кімнаті в котрій я живу. Для покращення результатів та демонстрування можливостей нейронної мережі до знаходження складних закономірностей, я створював важкі умови, такі як відкривання вікна для різкого зниження температури, підвищував температуру у кімнаті, збільшував та зменшув вологість та добавив складову шуму до даних, як спосіб аугментації даних, що допоможе проти перенавчання.

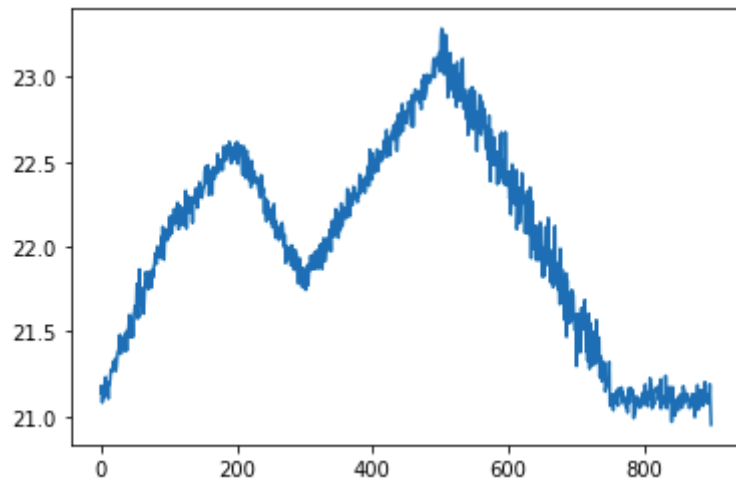


Рис.5.2.1. Приклад зібраних температурних даних

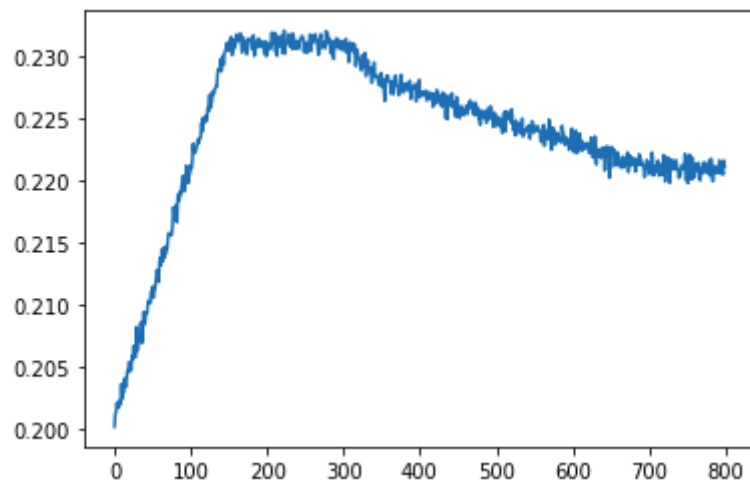


Рис.5.2.2. Приклад зібраних даних вологості приміщення

На рис. 5.2.1. та рис. 5.2.2 показані деякі числові дані зі зібраного датасету, який буде використовуватися для навчання та тестування системи. Ці графіки - лише деяка частина даних, так як не бачу сенсу будувати графіки всього датасету.

5.3. Етап навчання

Для навчання температурні дані були нормалізовані до проміжку від 0 до 1.

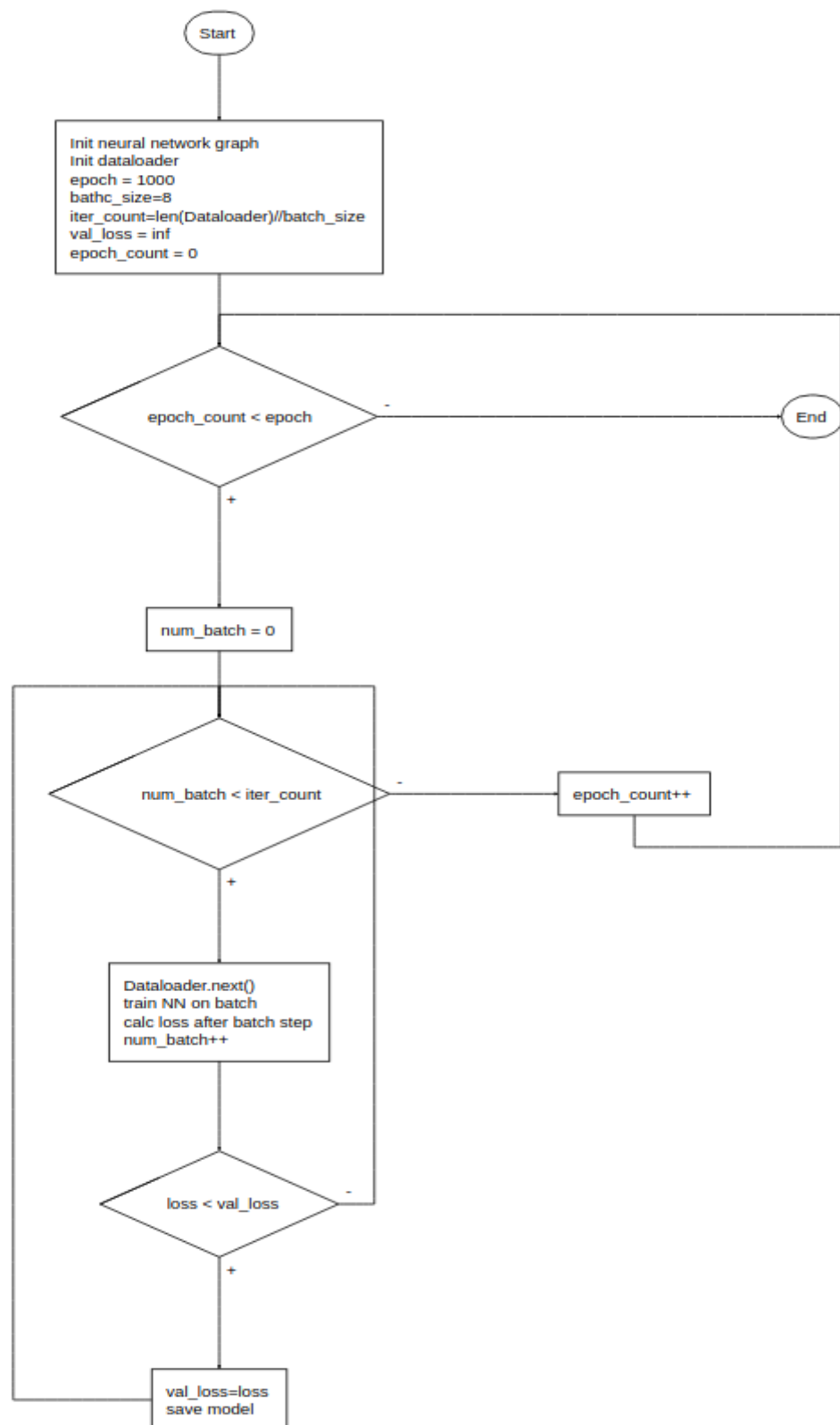


Рис.5.3.1. Блок схема коду для навчання нейронної мережі

Код для навчання відображен блок схемою рис. 5.3.1.

Процес навчання нашої нейронної мережі можна описати так:

1. Спочатку відбувається ініціалізація розрахункового графу нейронної мережі та ініціалізуються випадково ваги нашої моделі. Також на цьому етапі ми ініціалізуємо оптимізатор, в цьому випадку я буду використовувати Adam, зі стандартними параметрами, та швидкістю навчання 0,001
2. Ініціалізується екземпляр класу Dataloader, який буде відповідати за формування даних для навчання.
3. Ініціалізуємо гіперпараметри, а саме: кількість епох для навчання, розмір пакету для навчання
4. Вираховуємо кількість навчальних ітерацій у межах однієї епохи.
5. Ініціалізуємо тестувальний лосс великим значенням та ініціалізуємо значення кількості епох нулем.
6. Стартує навчання при якому ініціалізується кількість пакетів нулем і порівнюється значення з кількістю ітерацій, якщо кількість пакетів більша за кількість ітерацій, значить, що епоха пройшла, ми інкрементуємо значення кількості епох переходимо до цього ж пункту. Якщо менша, то переходимо до наступного пункту
7. Отримуємо тестувальний та тренувальний пакети, проводимо навчання на тренувальному пакеті, проводимо тестування на тестувальному пакеті. Інкримінуємо значення кількості ітерацій. Підраховуємо значення функції помилки на тестувальному пакеті, якщо вона менша ніж ініційований лосс, то призначаємо це значення лосу на тестуванні. Та зберігаємо модель.

В якості метрики навчання, буде використовуватися середньоквадратична функція помилки. Її ми й будемо мінімізувати. В якості оптимізатора буде використовуватися Adam, так як в даний момент він являється одним із кращих. Початкова швидкість навчання дорівнюватиме 0,001.

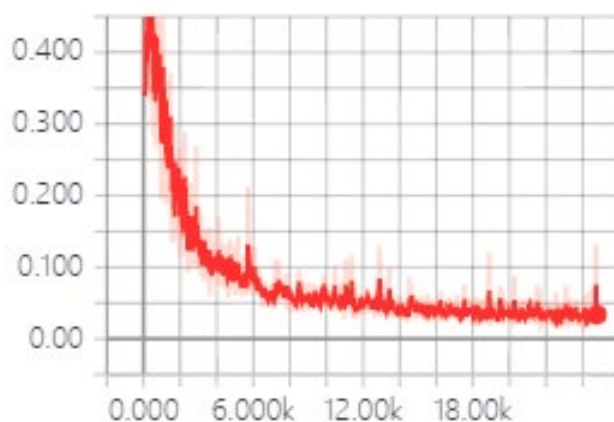


Рис 5.3.1. Графік функції помилки при навчанні нейронної мережі для прогнозування температури

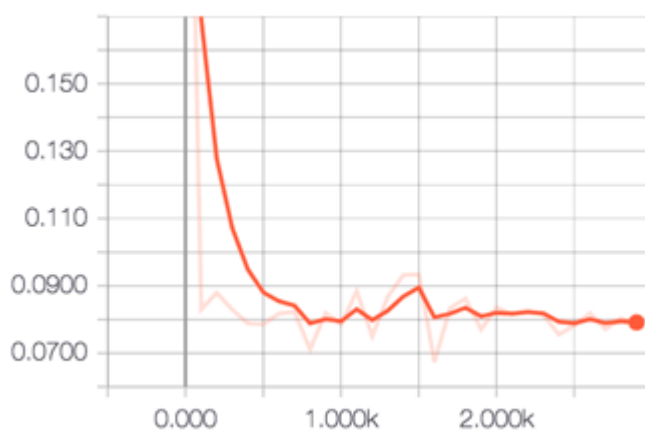


Рис. 5.3.2. Графік функції помилки при навчанні нейронної мережі для прогнозування вологості

Як можна побачити з рис. 5.3.1. та 5.3.2, навчання пройшло успішно, функції помилки зійшлись. З них видно, що для нейронної мережі знаходження часової залежності було легше ніж для температурної. Цей результат був досить очікуваних, виходячи з тих графіків, котрі я зобразив на рис.5.2.1, рис.5.2.2.

6. Архітектура прототипу інтелектуальної системи датчиків

6.1 Архітектура серверу

При проектуванні серверу, для роботи з сокетом використовувався асинхронний підхід побудований на файлових дескрипторах. Це дало змогу не блокувати роботу програми при прослуховуванні портів, та виконувати потрібні підпрограми, а саме, обчислення прогнозування нейронної мережі, побудова графіків, тощо.

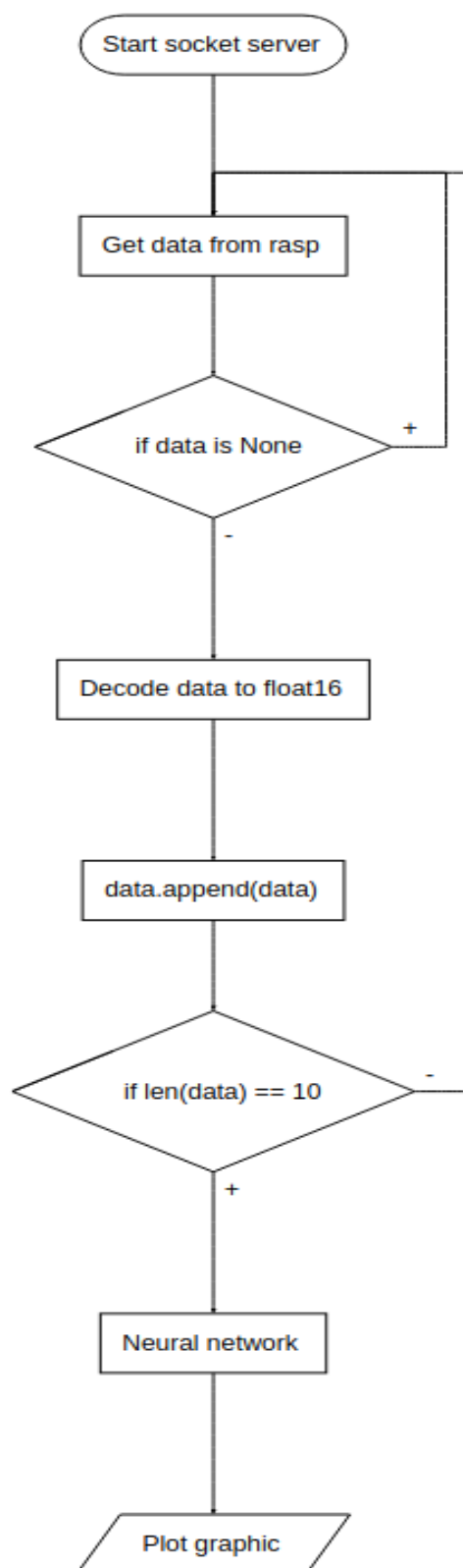


Рис. 6.1.1 Блок схема коду сервера

На рис. 6.1.1. зображена блок схема коду роботи серверу. На етапі "start socket server" відбувається ініціалізація об'єкту класу socket, прив'язування до

відповідного внутрішнього IP-адресу, який знаходиться за допомогою bash скрипту, який шукає у внутрішній локальній мережі Raspberry Pi, та порту, що задається у файлі конфігурації, та початок слухання цього порту.

6.2. Архітектура клієнта

Клієнтом виступає Raspberry Pi, до інтерфейсу вводу та виводу загального призначення підключений сенсор температури.

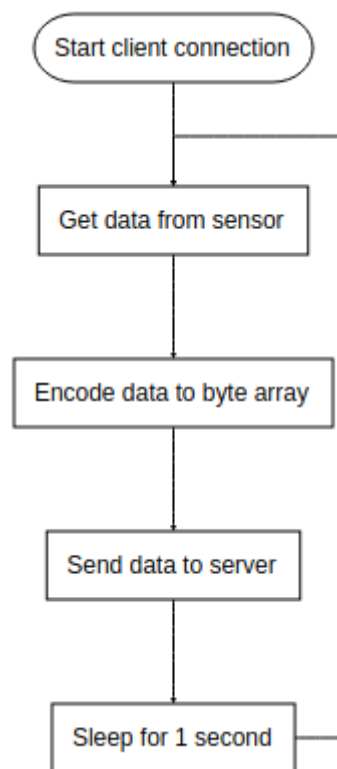


Рис. 6.2.1. Блок-схема коду клієнта

Спочатку отримуються дані з сенсору, за допомогою програми, яку розповсюджує виробник, після цього, дані кодуються до байтового масиву та відправляються до серверу. Після цього робота коду призупиняється на 2 секунди, як того вимагає виробник. Все це видно на рис 5.2.1.

7. Збір інтелектуальної системи

7.1. Підключення сенсору DHT22 до Raspberry Pi

Інформаційний вивід сенсору DHT22 підключається до Raspberry Pi за допомогою портів загального вводу-виводу, живлення під'єднується до першого порту, з якого подається напруга 3,3В, земля під'єднується до 9-го порта.

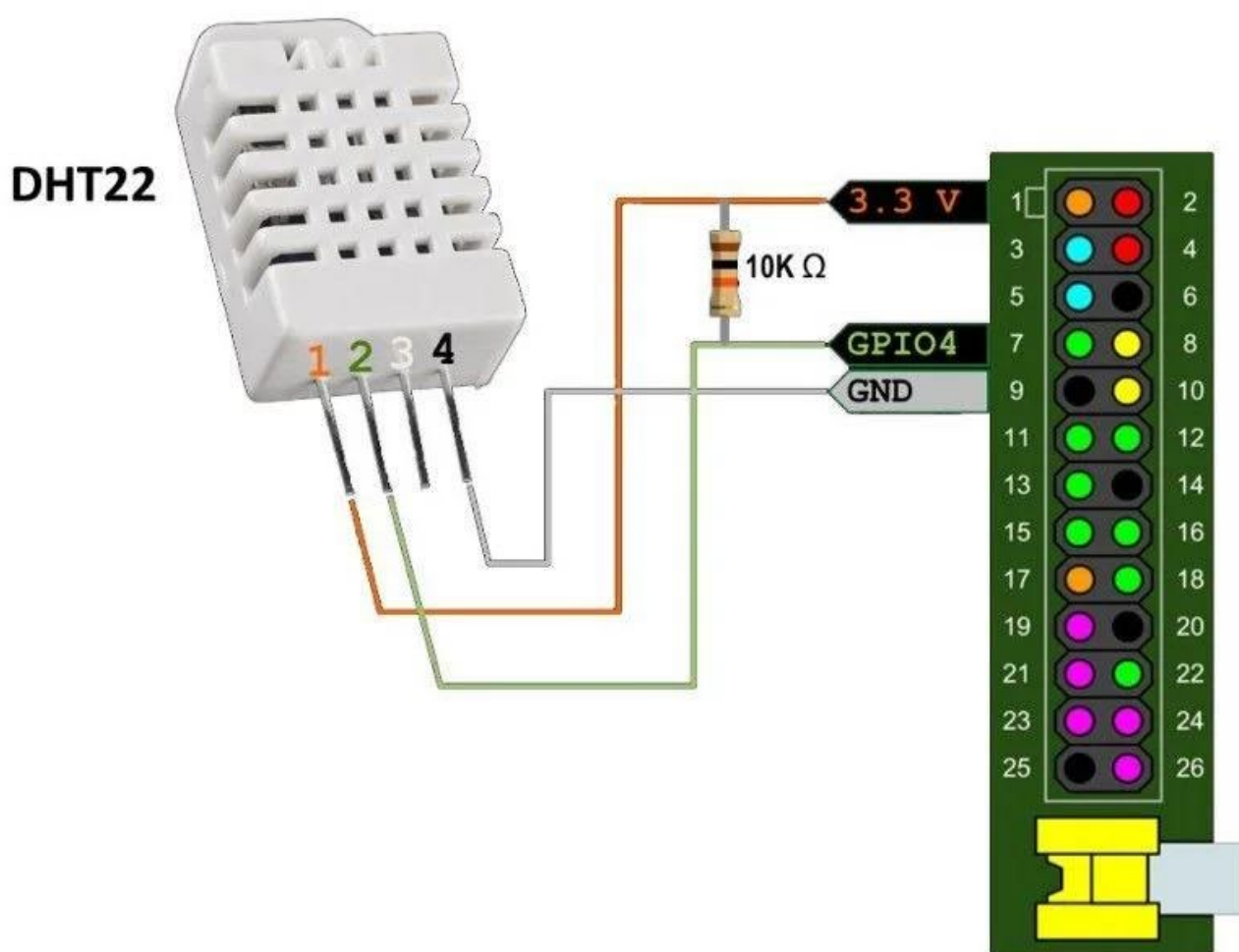


Рис.7.1. Схема підключення DHT22 до Raspberry Pi

Також ця схема потребує підтягуючий резистор номіналом 10 кОм. Він потрібен для отримання коректних даних з інформаційного виходу. Сигнал з інформаційного виходу кодується у цифровий.

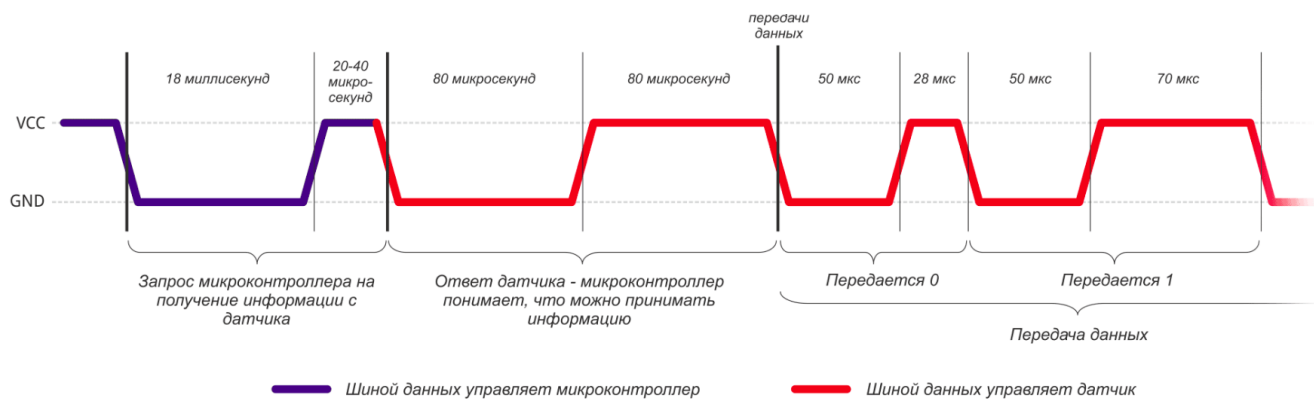


Рис.7.1.2. Опис сигналу з інформаційного виходу.

Описати принцип комунікації можна так:

1. Мікроконтролер подає стартовий сигнал - притискає шину до землі на 18-20 мс;
2. Після цього контролер відпускає лінію і стежить за рівнем на ній, приблизно 20-40 мкс;
3. Датчик, виявивши сигнал і почекавши поки рівень знову стане високим, сам притискає шину до землі на 80 мкс, за цей час робляться вимірювання і перетворення результатів;
4. Потім DHT22 відпускає лінію на 80 мкс, що вказує на те, що він готовий відправити дані;
5. Потім він відправляє 40 біт даних. Перед відправкою кожного біта датчик притискає шину до землі на 50 мкс, за яким слід 26-28 мкс для «0» або 70 мкс для «1»;
6. По завершенні зв'язку лінія витягується підтягуючим резистором і переходить в стан очікування.

Коли датчик вологості і температури відправляє дані, він спочатку відправляє MSb (Most Significant Bit) - старший значущий біт. Дані від датчика передаються у вигляді посилки, що складається з 40 біт даних - це 5 байт з яких перших два вологість, наступні 2 температура і байт парності. Байт парності дорівнює сумі попередніх байт. 1 і 2 байти містять відповідно цілу і дробову

частину інформації про вологість, 3 і 4 байт містять цілу і дробову частину інформації про температуру.

Таблиця. 7.1.1. Опис байтів у закодованому сигналі

Номер байту	Значення
1-ий	відносна вологість - ціла частина в%
2-ий	десята частина відносної вологості в%
3-ій	ціла частина температури в ° C
4-ий	десята частина температури в ° C
5-ий	контрольна сума (останні 8 біт {1-й байт + 2-й байт + 3-й байт + 4-й байт})

Для розкодування сигналу з DHT22 буде використовуватися бібліотека з відкритим доступом Ada_fruit.

```
pi@raspberrypi:~ $ cd Adafruit_Python_DHT/
pi@raspberrypi:~/Adafruit_Python_DHT $ sudo python examples/AdafruitDHT.py 22 18
Temp=21.7* Humidity=50.0%
pi@raspberrypi:~/Adafruit_Python_DHT $ □
```

Рис.7.1.3. Перевірка роботи бібліотеки Ada_fruit

На рис.7.1.3. зображен приклад роботи готової програми під Python. Як можна побачити все працює коректно. Для підключення сенсору треба вказати номер моделі, тобто 22, там номер порту, в даному випадку датчик був підключений до 18-го порту спільного використання.



Рис.7.1.3. Інтелектуальна система у підключеному вигляді

Прототип інтелектуальної системи зображен на рис.7.1.3. В цьому проекті використовується датчик DHT22 вже з підтягуючим резистором, використання його дозволило не витратити час на створення власної друкованої плати.

8. Розробка стартап проекту

8.1 Опис ідеї стартап проекту

Таблиця 8.1.1. Опис ідеї стартап проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Інтелектуальна система моніторингу та клімат контролю	1. Науковий, для контролю та прогнозування температурний та вологих параметрів приміщення	Є можливість планування експериментів, отримання інформації про майбутній стан кімнати, та можливість правильно планувати експеримент, виходя з цієї інформації
	2. Промисловий, для контролю та прогнозування температурний та вологих параметрів виробничих приміщень	Є можливість автоматизувати процеси виробництва, де необхідний контроль кліматичних параметрів приміщення, також є можливість передбачати можливі наслідки виходя із прогнозування системи
	3. Розумний дім, для контролю та прогнозування температурний та вологих параметрів житлових приміщень	Є можливість контролю оптимальних параметрів приміщення, за рахунок завчасної передачі інформації на інші модулі розумного дома, для підтримки оптимальних параметрів

Таблиця 8.1.2. Опис ідеї стартап проекту

№ п / п	Техніко-економічні характеристики ідеї	(потенційні) товари/концепції конкурентів				W (слабка сторона)	N (нейтральна сторона)	S (сильна сторона)
		Мій проект	Zipato	iqhouse	Nest			
1	Розміри та мобільність	Невеликі розміри, велика	Великі розміри, мала	Малі розміри, низька	Малі розміри, середня			+
2	Складність налаштування	Просте	Середнє	Складне	Просте		+	
3	Ціна	Низька	Велика	Велика	Середня			+
4	Мультивимірність	В наявності	Ні	Ні	Ні			+
5	Прогнозування	В наявності	Ні	Ні	Ні			+

8.2. Технологічний аудит ідеї проекту

Таблиця 8.2.1. Технологічна здійсненність ідеї проекту

№ п/п	Ідея проекту	Технології реалізації	Наявність технологій	Доступність технологій
		Електричні компоненти	Наявні	Доступні
		Датчик температури	Наявні	Доступні
		Датчик вологості	Наявні	Доступні
		Одноплатний комп'ютер	Наявні	Доступні

		Нейронні мережі	Наявні	Доступні
Обрана технологія реалізації ідеї стартап проекту. Всі компоненти наявні в широкому доступі				

8.3. Аналіз ринкових можливостей запуску стартап-проекту

Таблиця 8.3.1. Попередня характеристика потенційного ринку стартап-проекту

№, п/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	10
2	Загальний обсяг продаж, грн/ум.од	~700 мільйонів
3	Динаміка ринку (якісна оцінка)	Зростає
3	Наявність обмежень для входу(вказати характер обмежень)	Відсутні
4	Специфічні вимоги до стандартизації та сертифікації	Відсутні
5	Середня норма рентабельності в галузі(або по ринку), %	~10%

Таблиця 8.3.2. Характеристика потенційних клієнтів стартап-проекту

№, п/п	Потреба що формує ринок	Цільова аудиторія(цілові сегменти ринку)	Відмінності у поведінці різний потенційних цільових груп клієнтів	Вимоги споживачів до товару
1	Автоматизація процесів навчання та дослідження та контроль досліджувальн	Науковці, дослідники, учні, педагоги, ентузіасти, студенти	Поведінку клієнта формують: мала ціна, швидкість доставки,	Якість роботи кінцевого продукту, його низька ціна

	ого середовища		зручність у користуванні, мобільність приладу, мультизадачність	
2	Автоматизація та контроль виробничих процесів	Підприємці та виробники також робітники цих підприємств	Поведінку клієнта формують: надійність, мала ціна, швидкість доставки, зручність у користуванні, мобільність приладу, мультизадачність, здатність до прогнозування	Якість роботи кінцевого продукту, можливість до прогнозування , швидкість доставки та підтримка з боку виробника
3	Використання в якості кліматичного модуля в системах розумного дому	Будь-хто	Поведінку клієнта формують: мала ціна, швидкість доставки, зручність у користуванні, мобільність приладу, мультизадачність, здатність до прогнозування	Якість роботи кінцевого продукту, простота в користуванні, мала ціна, інтелектуальність, підтримка з боку виробника

Таблиця 8.3.3 Фактори загроз

№,	Фактор	Зміст загрози	Можлива реакція
----	--------	---------------	-----------------

п/п			компанії
1	Потрібні кваліфіковані кадри	Потреба в людях з певними освітніми навиками	Рекрутинг робочого персоналу в науководослідних інститутах
2	Ресурсна проблема	Проблема у виготовленні кінцевого продукту	Укладання договорів з державними структурами для фінансування та надання можливостей для пошуку нових постачальників
3	Глобальна економічна криза	Проблеми з капіталізацією, зниження рівня продажів	Підвищення ціни на продукт. Розробка більш дешевого пристрою, пошук нових ринків збуту
4	Відсутність вільного капіталу	Проблеми з новими розробками та розрахунками	Отримання банківських кредитів, продаж акцій, пошук нових інвесторів, ІСО
5	Проблеми постачання	Проблеми з виготовленням нових пристроїв та підтримкою старих	Пошук нових постачальників комплектуючих, розробки нових систем із інших компонентів

Таблиця 8.3.4. Фактори можливостей

№,	Фактор	Зміст можливості	Можлива реакція
----	--------	------------------	-----------------

п/п			компанії
1	Конкуренція	Винайдення та розробка нового товару з більш кращими властивостями, зниження цін	Удосконалення власного товару шляхом підвищення якості роботи та зниження ціни кінцевого продукту
2	Попит	На ринку має місце велика зацікавленість в таких інноваційних приладах з боку звичайних користувачів та виробників	Підвищення жорсткої рекламної діяльності, просування товару в мережі інтернет та офлайн
3	Зростання ринку	Зростання об'ємів продаж	Накопичення капіталу компанії для подальшого розвитку продукту та розширення штату

Таблиця 8.3.5 Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1. Вказати тип конкуренції - олігополія	На ринку присутня не велика кількість компаній, які займаються розробкою систем клімат контролю та модулів для систем розумного дому	Наймати більш кваліфікованих кадрів для виготовлення більш якісного продукту, використання передових технологій та зниження ціни продукції.
2. За рівнем конкурентної боротьби	Місце знаходження компаній не обмежується	Створення офісів підтримки клієнтів в

-національний	територіально; офіси фірм розміщуються у різних містах та країнах	різних країнах, налагоджування розповсюдження продукції компанії у різні країни, адаптація реклами під різні країни
3. За галузевою ознакою - внутрішньогалузева	Економічна конкуренція між різними товаровиробниками систем які працюють в одній галузі виробництва, створюють та продають однакові пристрої, що звільнятимуть одну й ту ж саму потребу споживачів, однак мають деякі відмінності у затратах на виробництво, якості роботи та ціні продукту.	Дослідження пристроїв конкурентів та швидка реакція на покращення якості, або зниження ціни на товари конкурентних компаній
4. Конкуренція за видами товарів: -товаро-видова	Конкуренція між компаніями, які виробляють продукт одного виду	Знижувати ціну товару, покращувати якість роботи, створювати програми лояльності для нових клієнтів
5. За характером конкурентних переваг -нецінова	Потреба в виграванні на ринку серед конкурентів за рахунок якості, кількості та функціоналу кінцевого продукту	Покращення якості роботи кінцевого продукту, здешевлення ціни виробництва та, звідси, здешевлення ціни кінцевого продукту, збільшення кількості товару на ринку, розширення функціоналу,
6. За інтенсивністю -марочна	На ринку присутні компанії, котрі відомий та гучний бренд	Роботи свій бренд більш впізнаваним серед споживачів та у Світі в цілому.

Таблиця 8.3.6 Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
	Zipato IqHouse Nest	-	Кількість та якість постачальників	Низька ціна, надійність, швидка доставка, зручність, здатність до прогнозування і адаптації, зручність користування, мобільність, широкий спектр застосування	Замінники існують, проте вони не мають змогу до прогнозування і адаптації
Висновки:	Інтенсивність висока, так як товар є складним у виготовленні та підтримки	-	Постачальники мають великий вплив так як конкуренція має національні властивості, а також велике значення має швидкість між виготовленням та клієнтом		

Таблиця 8.3.7. Обґрунтування факторів конкурентоспроможності

№, п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників що роблять фактор для порівняння конкурентних проектів значущим)
--------	-------------------------------	--

1	Ціна	Ціна нижча ніж у конкурентів
2	Масовість виробництва та терміни	Зниження за часом ціни кінцевого продукту та вчасні поставки продукції
3	Достатня якість продукції	Більше гнучкі можливості а ніж у конкурентів

Таблиця 8.3.8 Порівняльний аналіз сильних та слабких сторін інтелектуальної системи моніторингу та клімат контролю

№, п/п	Фактор конкурентноспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з ControlAI						
			-3	-2	-1	0	+1	+2	+3
1	Ціна	14		+					
2	Терміни та масовість виробництва	15			+				
3	Інтелектуальність	20	+						
4	Мобільність	19		+					
5	Мультипараметричність	20	+						

Таблиця 8.3.9 SWOT - аналіз стартап-проекту

Сильні сторони: Гнучкість використання, здатність до прогнозування, мобільність, дешевизна виготовлення, надійність, простота у	Слабкі сторони: масовість виробництва у порівнянні з конкурентами, незнайомий бренд, відсутність екосистеми відомих брендів
---	---

використанні	
Можливості: розвивати інтелектуальну частину проекту, підвищувати якість продукту, робити бренду ім'я на ринку	Загрози: поява нових конкурентів; нестача кваліфікованих кадрів на ринку праці, можливе зниження попиту, економічна криза

Таблиця 8.3.10 Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Розробка більшого функціоналу інтелектуальної частини	60%	8 місяців
2	Розробка інших частин інтелектуальної системи моніторингу та клімат контролю	40%	3 місяці

8.4. Розроблення ринкової стратегії стартап проекту

Таблиця 3.4.1 Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової	Готовність споживачів сприйняти	Орієнтовний попит в межах	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
-------	-----------------------	---------------------------------	---------------------------	--------------------------------------	--------------------------

	групи потенційних клієнтів	продукт	цільової групи (сегменту)		
1	Дослідницькі лабораторії	готові	Високий	Середня	Висока
2	Навчальні заклади	готові	Низький	Середня	Низька
3	Промисловість	готові	Високий	Висока	Низька
4	Очиничні клієнти	готові	Високий	Висока	Середня
Які цільові групи обрано: Дослідницькі лабораторії, навчальні заклади, промисловість, очиничні клієнти					

Таблиця 8.4.2. Визначення базової стратегії розвитку

№ п/п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку*
1	Розробка більшого функціоналу інтелектуальної системи моніторингу та клімат контролю	Розширення ринку за рахунок вирішення більшого спектру задач та потреб кінцевих користувачів	Вирішується більше проблем та задач кінцевих користувачів	Стратегія лідерства на витратах
2	Розробка інших частин інтелектуальної системи моніторингу та клімат контролю	Розширення ринку за рахунок вирішення більшого спектру задач	Вирішується більше проблем та задач кінцевих користувачів	Стратегія лідерства на витратах

		та потреб кінцевих користувачів		
--	--	---------------------------------------	--	--

Таблиця 8.4.3. Визначення базової стратегії конкурентної поведінки

<i>№ п/ п</i>	<i>Чи є проект «першопрохідцем » на ринку?</i>	<i>Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?</i>	<i>Чи буде компанія копіювати основні характеристики товару конкурента, і які?</i>	<i>Стратегія конкурентної поведінки*</i>
1	Ні	Обидва варіанти	Так. Деякий функціонал	Заняття конкурентної ніші

Таблиця 8.4.4. Визначення стратегії позиціювання

<i>№ п/ п</i>	<i>Вимоги до товару цільової аудиторії</i>	<i>Базова стратегія розвитку</i>	<i>Ключові конкурентоспромо жні позиції власного стартап- проекту</i>	<i>Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)</i>
1	Ціна	Лідерства на витратах	Низька ціна	Оптимальне співвідношення ціна/якість, покращення існуючих характеристик.
2	Якість	Лідерства на	Висока якість	

		витратах		
3	Мобільність	Лідерства на витратах	Висока мобільність	
4	Точність	Лідерства на витратах	Висока точність	

8.5. Розроблення маркетингової програми стартап-проекту

Таблиця 8.5.1. Визначення ключових переваг концепції потенційного товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1	Низька ціна	Найнижча ціна на ринку	Найнижча ціна на ринку
2	Висока якість	Висока якість	Висока якість
3	Гнучкість використанн я	Можливість пристосуватис я до кліматичних умов користувачів	Можливість пристосуватися до кліматичних умов користувачів

Таблиця 8.5.2. Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові		
I. Товар за здумом	Інтелектуальна система моніторингу та клімат контролю пріміщень		
II. Товар у реальному виконанні	Властивості/характеристики	М/Нм	Вр/Тх /Тл/Е/Ор
	1. Швидкість роботи	М	ВР
	2. Точність вимірювання	М	Тх

	3. Споживана потужність	М	Тл
	Якість: відповідає вимогам стандартів		
	Пакування: пристрій, інструкція, кабель для зарядки		
	Марка: “HomeAI”		
III. Товар із підкріпленням	До продажу: договір, гарантія.		
	Після продажу: доставка, підтримка.		
За рахунок чого потенційний товар буде захищено від копіювання: патентні права.			

Таблиця 8.5.3. Визначення меж встановлення ціни

№ п/п	Рівень цін на товари- замінники	Рівень цін на товари- аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	-	500\$	1200\$	200 - 250 \$

Таблиця 8.5.4. Формування системи збуту

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Найкраща якість та більша кількість функцій за нижчу ціну	Налагодження контакту з клієнтами, надання достовірної інформації про характеристики, настройка за потреби.	Канал першого рівня	Через посередників

Таблиця 8.5.5. Концепція маркетингових комунікацій

№ п/п	Специфіка поведінки	Канали комунікацій,	Ключові позиції,	Завдання рекламного	Концепція рекламного
----------	------------------------	------------------------	---------------------	------------------------	-------------------------

	цільових клієнтів	якими користуютьс я цільові клієнти	обрані для позиціонуван ня	повідомлення	звернення
	Вибір найкращого продукту	Прямі - канал комунікації, коли інформація передається безпосередн ьо від інформатора до інформовано ї особи	Якість, низька кількість браку, низька ціна.	Надання повної інформації про переваги виробу	Хороші характерист ики за низької ціни

Висновки

В ході магістерської дисертації були виконані такі поставлені задачі:

- розгляд варіантів реалізації на сучасних обчислювальних системах
- розгляд технологій бездротового зв'язку та протоколи бездротової передачі інформації
- був зроблений огляд штучних нейронних мереж, аналіз структури сучасних нейронних мереж та була обрана архітектура, котра може вирішити задачу
- збір даних для навчання штучної нейронної мережі з обраною архітектурою
- було проведено навчання штучної нейронної мережі на зібраних даних
- реалізована архітектура клієнту та серверу для обміну між ними інформацією

Як показав робочий прототип, реалізація такої системи можливо, вона добре прогнозує на тестовій виборці даних, котрі були зібрані під час виконання магістерської дисертації.

Список літератури.

1. TCP/IP Illustrated, Vol. 1: The Protocols / W. Richard Stevens // Addison-Wesley Professional. – 1994. – Vol. 6 – PP. 439–440
2. The TCP/IP Guide: A Comprehensive, Illustrated Internet Protocols Reference / Charles M. Kozierok // No Starch Press. – 2005. – Vol. 4 – PP. 300–301
3. UNIX Network Programming / Prentice Hall. – 1990. Vol. 4 – PP. 254–256
4. Pineda F.J. Generalization of Back-propagation to Recurrent Neural Networks // Physics Review Letters. – 1987. – Vol. 18. – P. 2229 – 2232.
5. Werbos P.J. Generalization of Back-propagation with Application to a Recurrent as Market Model // Neural Networks. – 1988. – Vol.1. – P. 339 – 356.
6. Williams R.J., Zipser D., Learning A. Algorithm for Continually Running Fully Recurrent Neural Networks // Neural Computation. – 1989. – Vol. 1. – P. 270 – 280.
7. Осовский С. Нейронные сети для обработки информации. – М.: Финансы и статистика, 2004. – 343 с.
8. Boden M. A Guide to recurrent neural networks and backpropagation // The DALLAS project. Report from the NUTEC-supported project AIS-8: Application of Data Analysis with Learning Systems, 1999–2001. A. Host (ed.), SICS Technical Report T2002:03, SICS. – Kista, Sweden, 2002.
9. Cypress Hits Half-Billion Mark in Shipments of PSoC® Programmable System-on-Chip Devices // <http://www.reuters.com/article/2009/03/10/idUS116335+10-Mar-2009+BW20090310>
10. PSoC family // <https://www.digikey.com/en/product-highlight/c/cypress/psoc-family>
11. Wi-Fi Secure MCU SoCs & Modules // https://www.redpinesignals.com/Products/wireless_secure_MCU/Wi-Fi_Secure_MCU_SoCs_&Modules/
12. What is Arduino // <https://www.arduino.cc/en/Guide/Introduction>
13. Compare board specs // <https://www.arduino.cc/en/products/compare>

14. RaspberryPI model comparison //

<http://socialcompare.com/en/comparison/raspberrypi-models-comparison>

15. The Large Text Compression Benchmark //

<http://www.matthmahoney.net/dc/text.html#1218>

16. <https://www.cypress.com/products/microcontrollers-mcus>

17. Why Tensorflow // <https://www.tensorflow.org/about/>